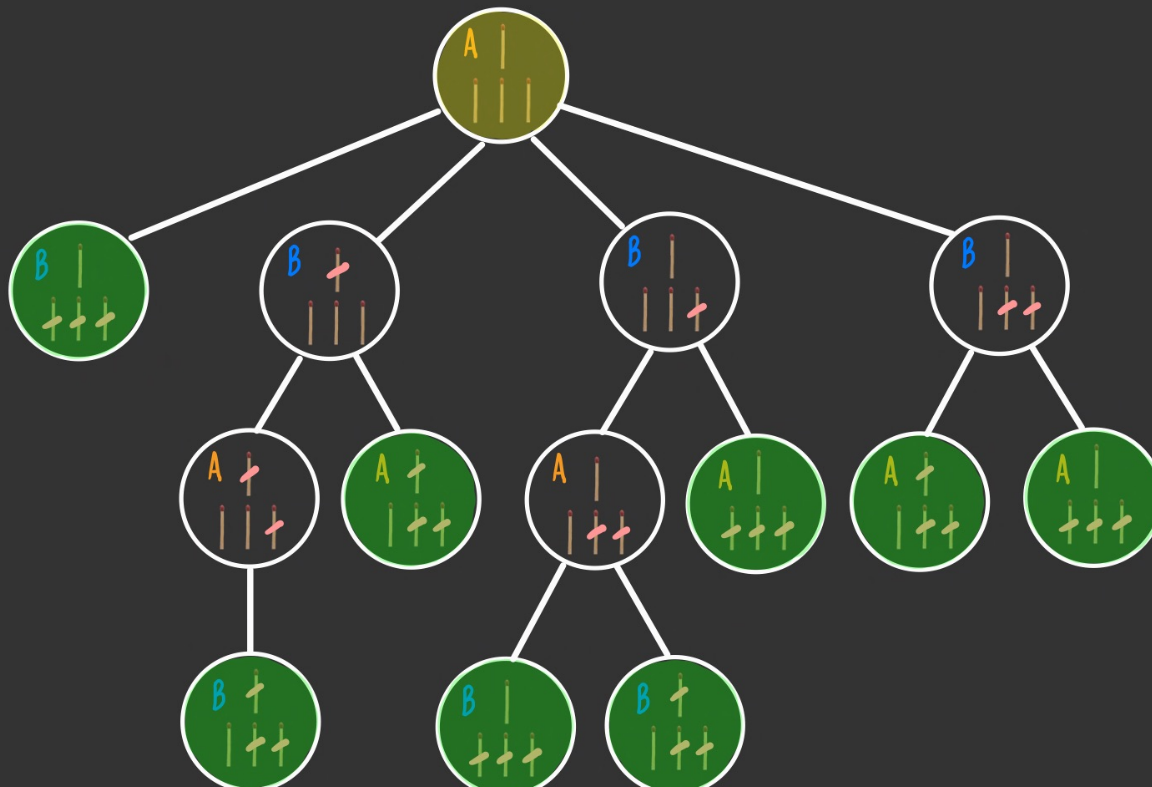
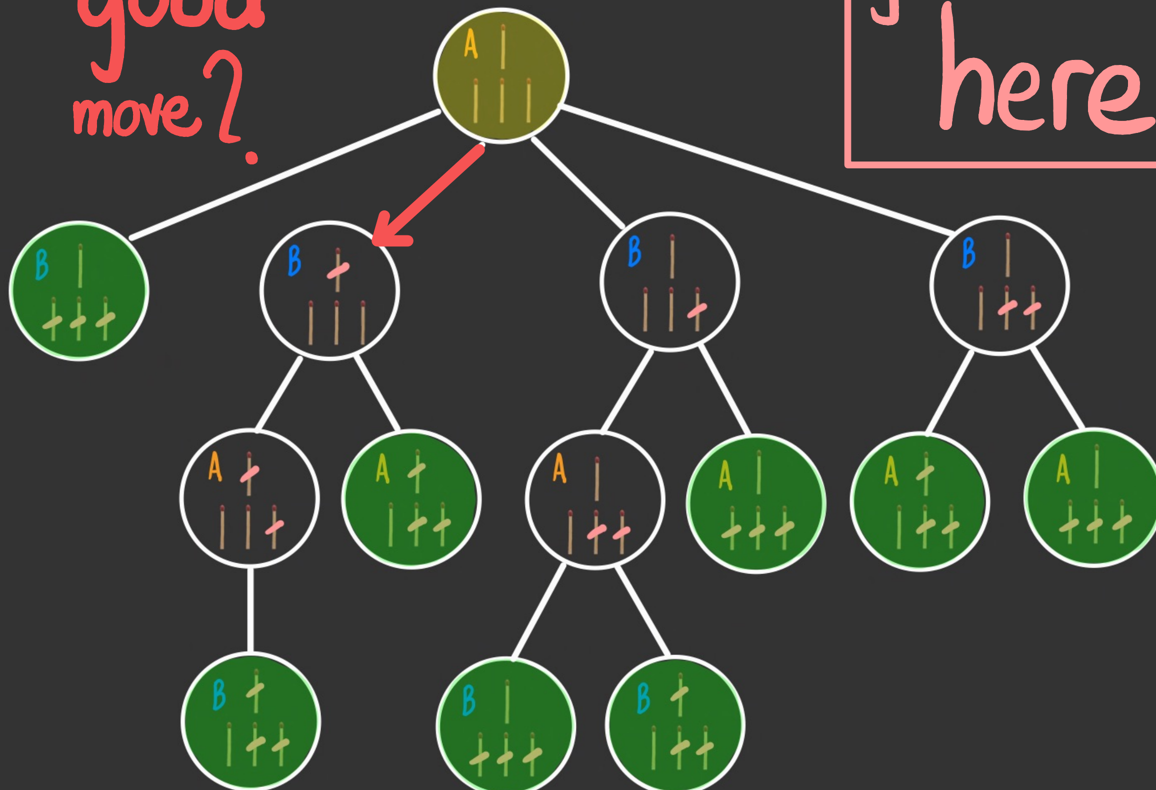
minimax

CSCI 373


A
B
B
B
B
A
A
A
A
A
A
B
B
B

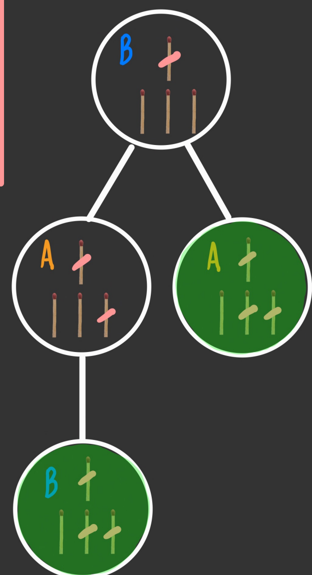
is this a good move?

your answer here



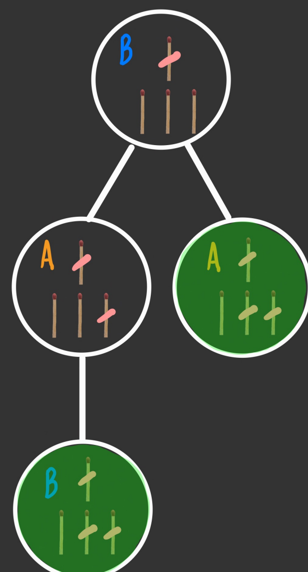
a **game** with players P is a tuple (M, U) where:

- $M = (Q, \Sigma, \Delta, q_0, F)$ is a state machine where each state $q \in Q$ has the form $(p, q') \in P \times Q'$ for an auxiliary set Q' of states
- utility function $U : F \times P \to \mathbb{R}$ which gives the value of each final state for each player

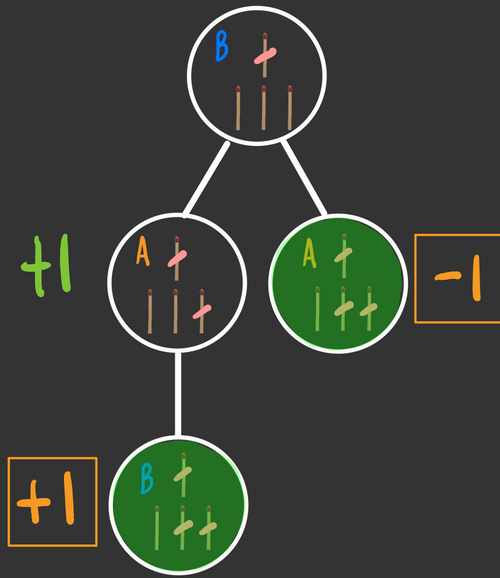


the minimax algorithm is based on the pessimistic principle that our opponent will always choose the move that is **worst** for us

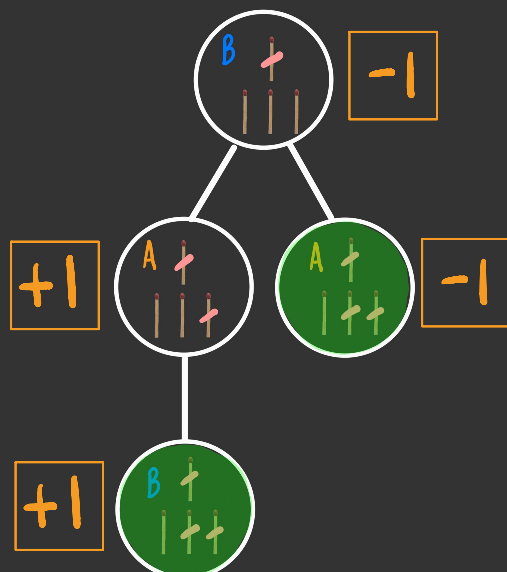
how might we say this more rigorously?



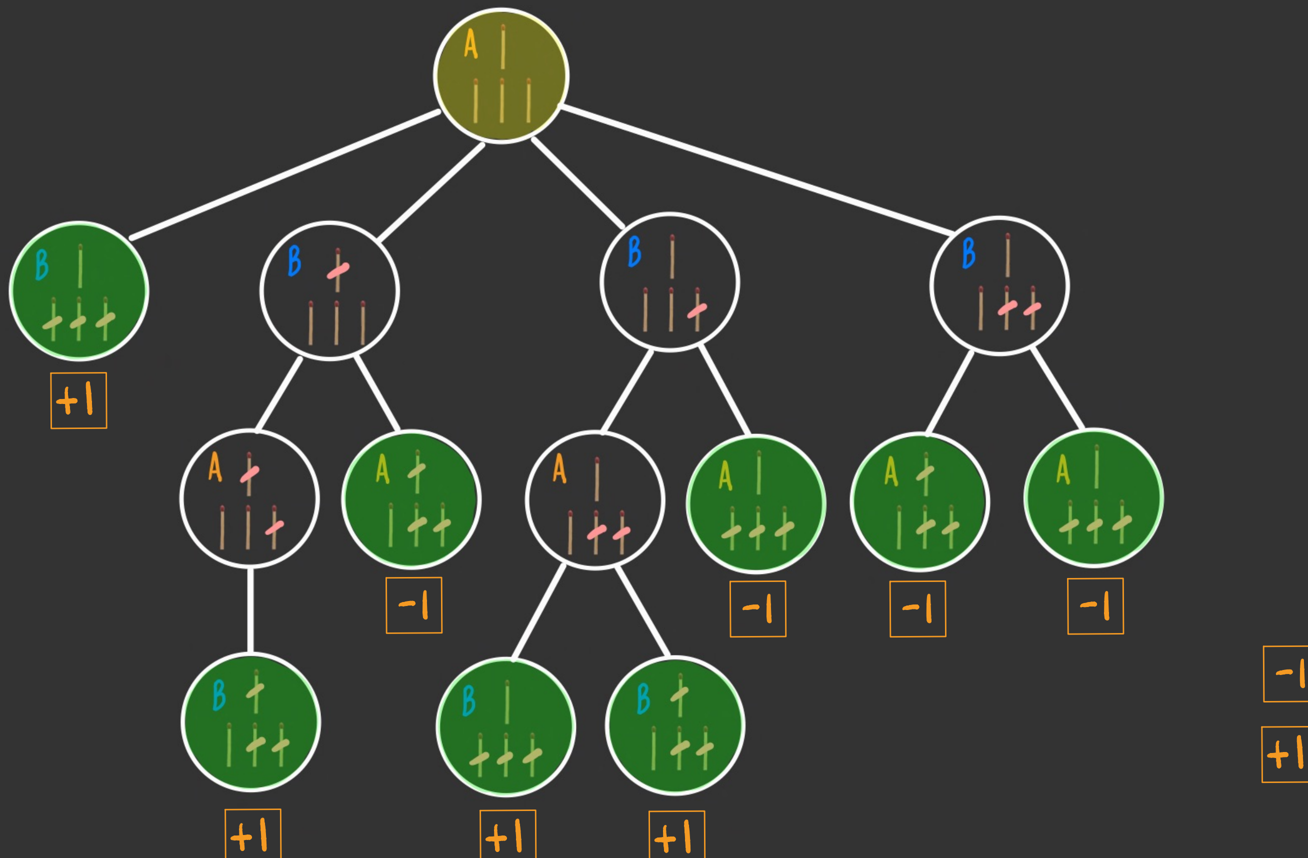
the minimax algorithm is based on the pessimistic principle that our opponent will always choose the move that gives us the lowest utility

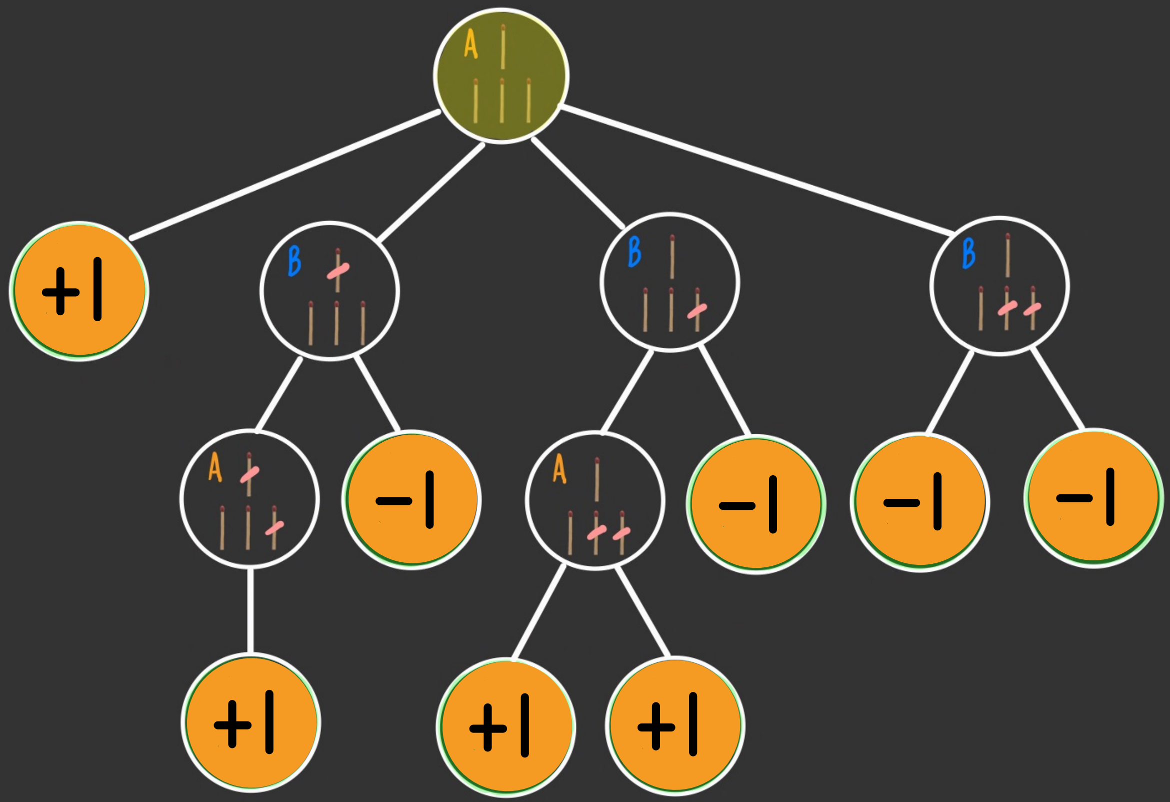


the minimax algorithm is based on the pessimistic principle that our opponent will always choose the move that gives us the lowest utility

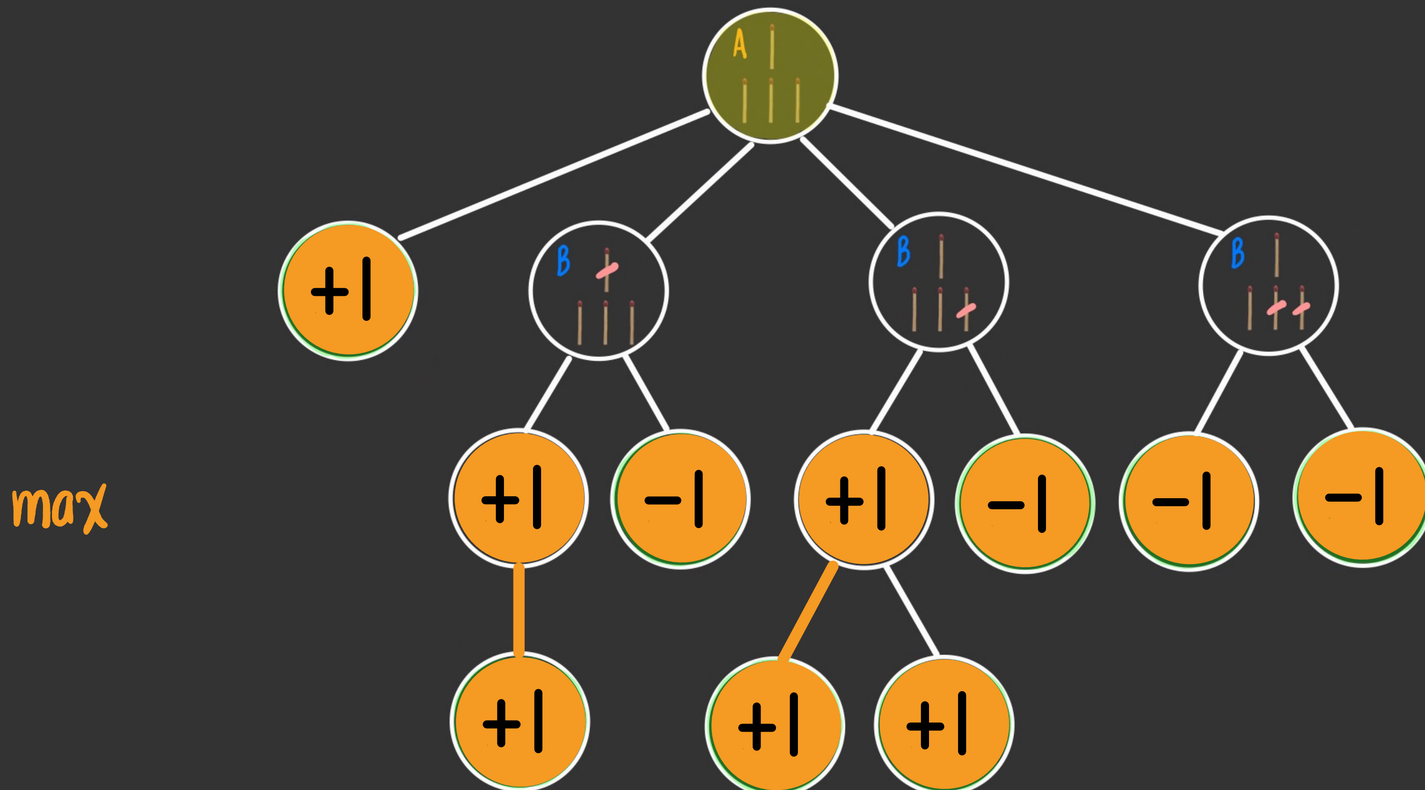


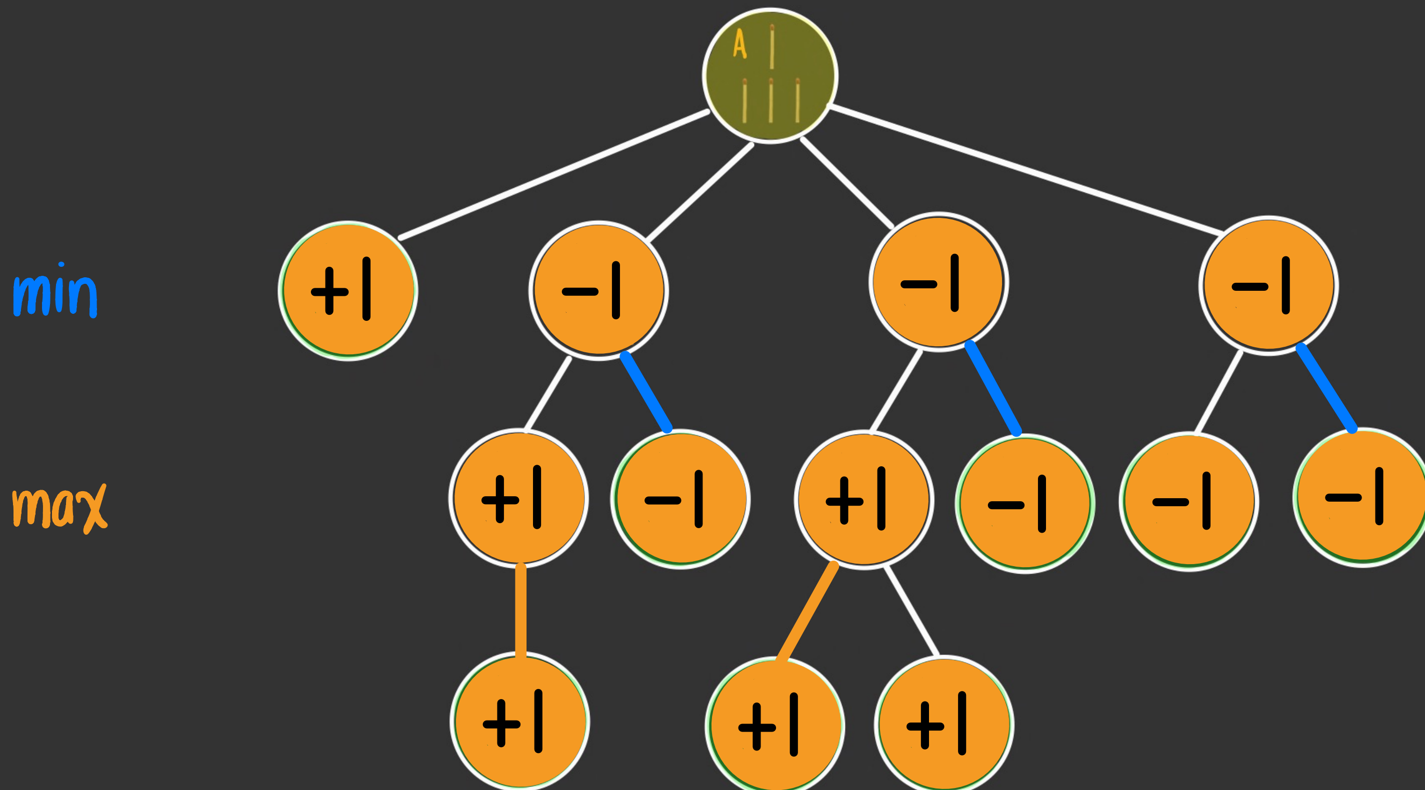
the minimax algorithm is based on the pessimistic principle that our opponent will always choose the move that gives us the lowest utility


A
B
B
B
B
+1
A
A
A
A
A
A
-1
-1
-1
-1
-1
B
B
B
+1
+1
+1
+1

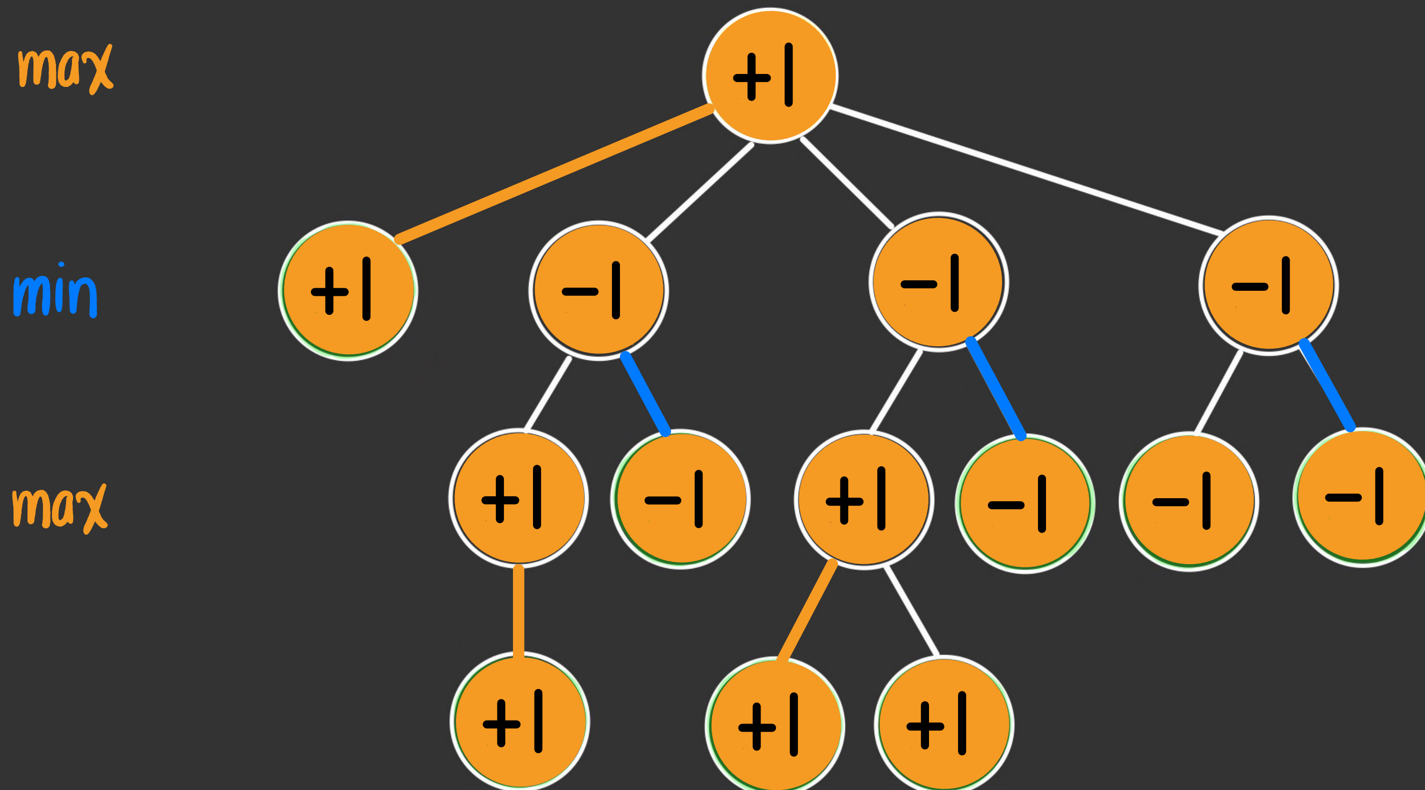


A
+1
B
B
B
A
-1
A
-1
-1
-1
+1
+1
+1


A
B
B
B
+1
+1
-1
+1
-1
-1
-1
+1
+1
+1
max

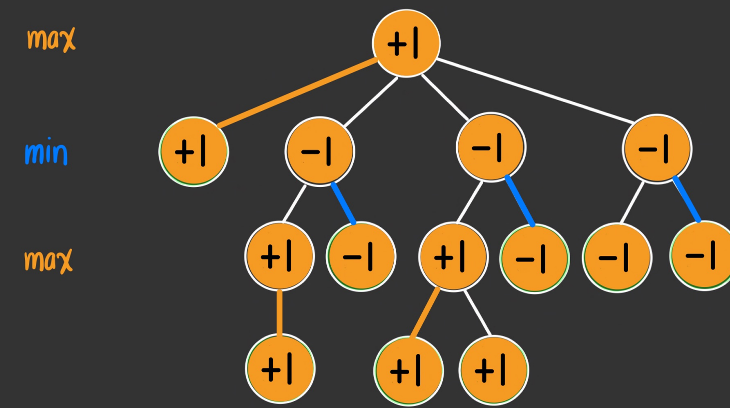


A
min
max
+1
-1
-1
-1
+1
-1
+1
-1
-1
-1
+1
+1
+1

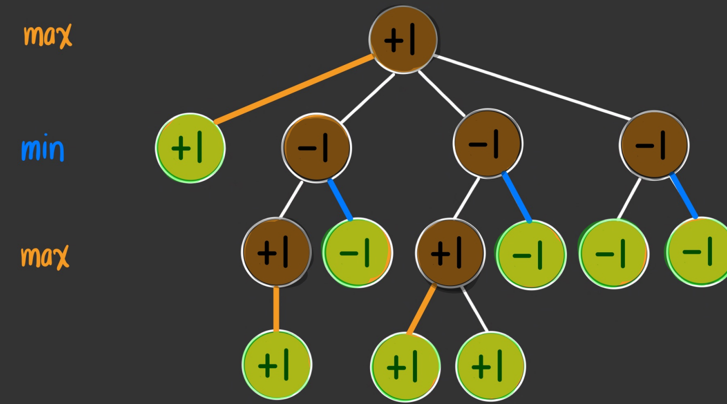


max
min
max
+1
+1
−1
−1
−1
+1
−1
+1
−1
−1
−1
+1
+1
+1

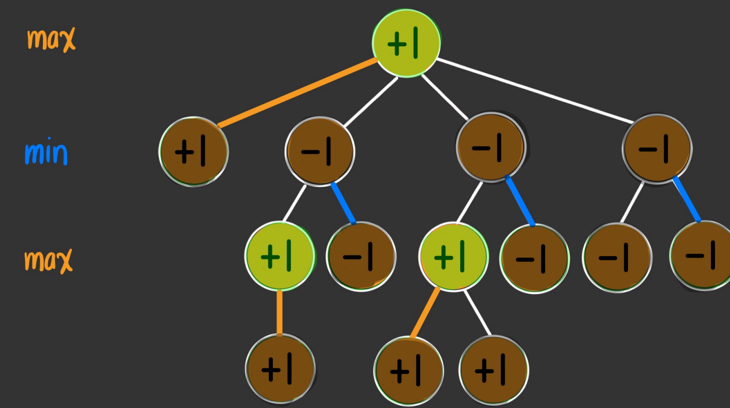
player whose utility we're interested in



$$\text{minimax}(q,p) = \begin{cases} U(q,p) & \text{if } q \in F \\ \max\limits_{q'}\left\{\text{minimax}(q',p) \;\middle|\; \langle q,\sigma,q'\rangle \in \Delta\right\} & \text{if } q \notin F \text{ and } p(q) = p \\ \min\limits_{q'}\left\{\text{minimax}(q',p) \;\middle|\; \langle q,\sigma,q'\rangle \in \Delta\right\} & \text{if } q \notin F \text{ and } p(q) \neq p \end{cases}$$



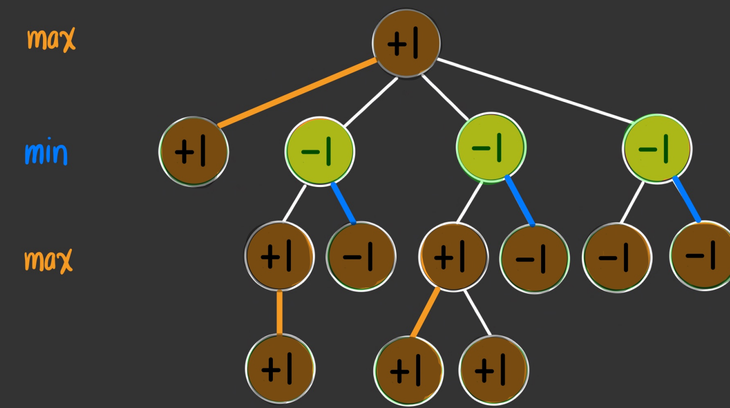
$$\mathrm{minimax}(q,p) = \begin{cases} U(q,p) & \text{if } q \in F \\ \max\limits_{q'} \left\{ \mathrm{minimax}(q',p) \,\middle|\, \langle q,\sigma,q' \rangle \in \Delta \right\} & \text{if } q \notin F \text{ and } p(q) = p \\ \max\limits_{q'} \left\{ \mathrm{minimax}(q',p) \,\middle|\, \langle q,\sigma,q' \rangle \in \Delta \right\} & \text{if } q \notin F \text{ and } p(q) \neq p \end{cases}$$



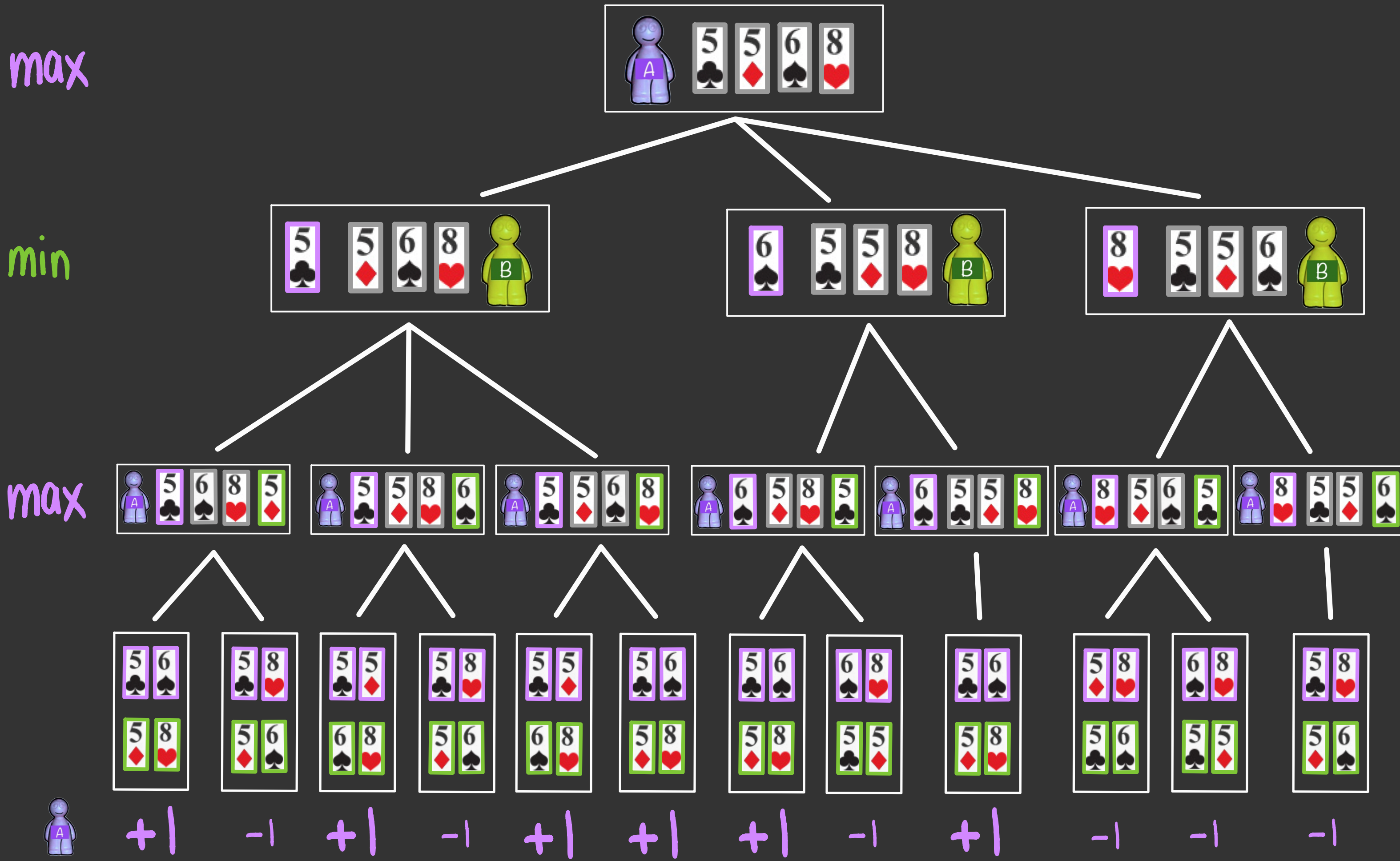
$$\text{minimax}(q,p) = \begin{cases} U(q,p) & \text{if } q \in F \\ \max\limits_{q'} \left\{ \text{minimax}(q',p) \,\middle|\, \langle q,\sigma,q' \rangle \in \Delta \right\} & \text{if } q \notin F \text{ and } p(q) = p \\ \min\limits_{q'} \left\{ \text{minimax}(q',p) \,\middle|\, \langle q,\sigma,q' \rangle \in \Delta \right\} & \text{if } q \notin F \text{ and } p(q) \neq p \end{cases}$$

player whose turn it is in state q

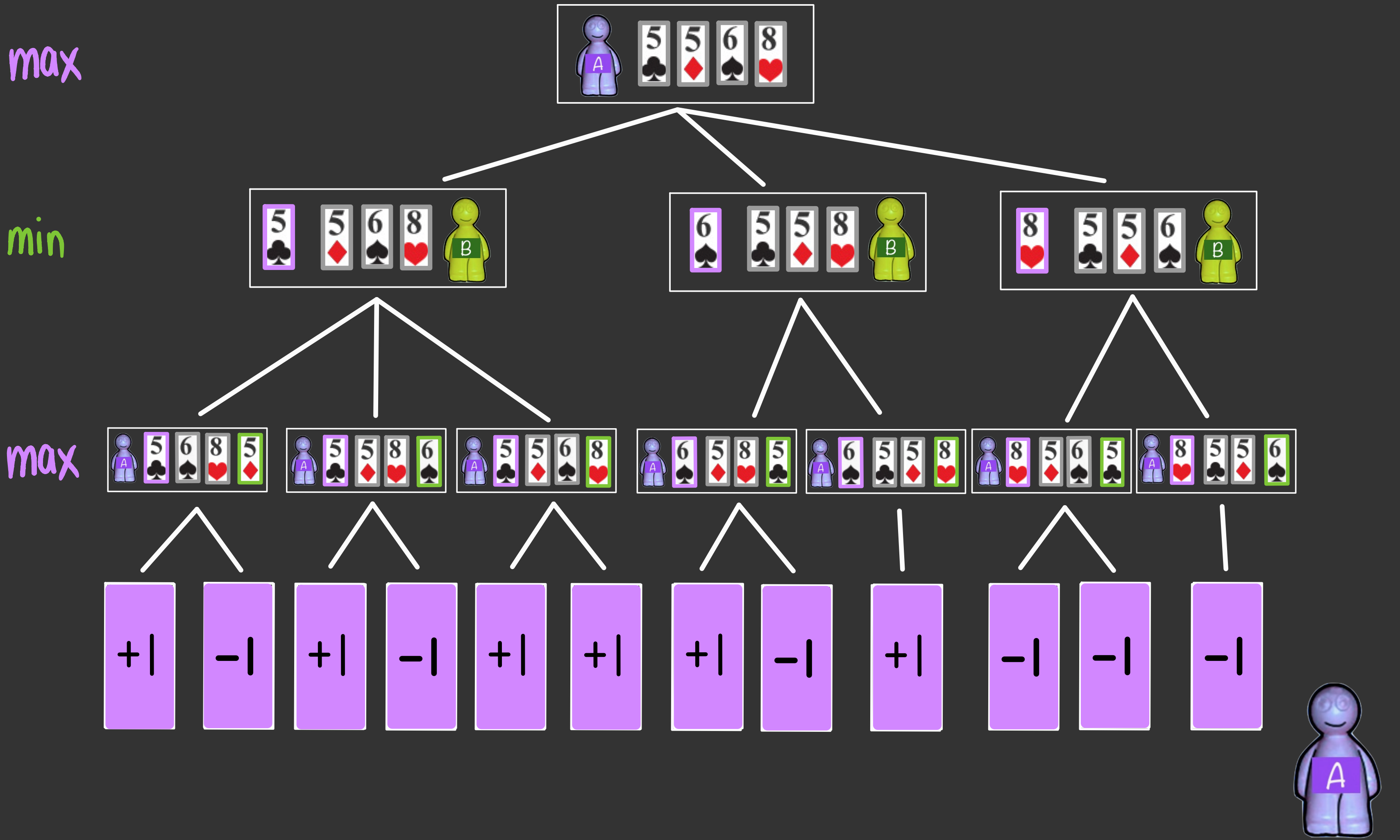
player whose utility we're interested in

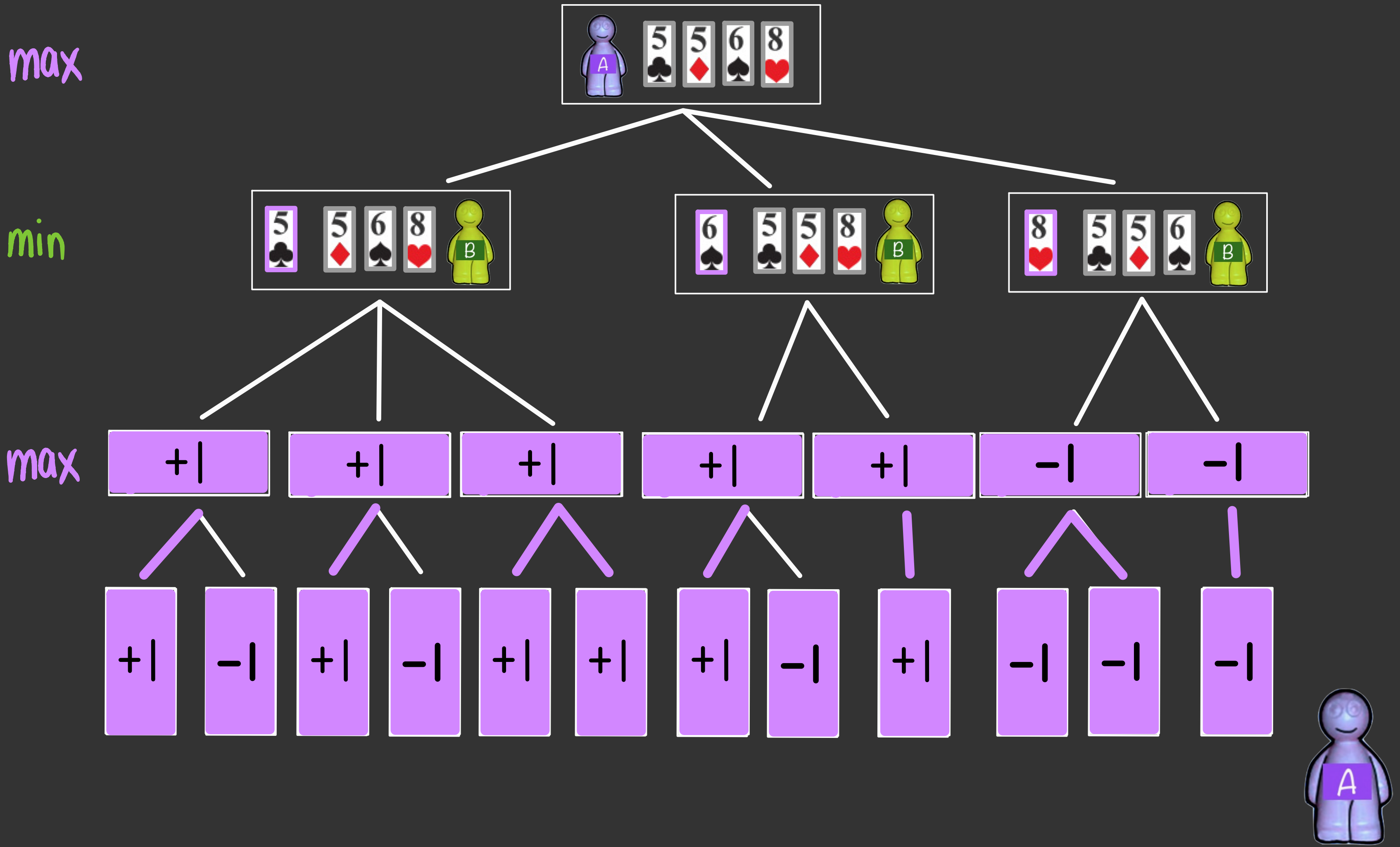


$$\mathrm{minimax}(q, p) = \begin{cases} U(q, p) & \text{if } q \in F \\ \max\limits_{q'} \left\{ \mathrm{minimax}(q', p) \;\middle|\; \langle q, \sigma, q' \rangle \in \Delta \right\} & \text{if } q \notin F \text{ and } p(q) = p \\ \min\limits_{q'} \left\{ \mathrm{minimax}(q', p) \;\middle|\; \langle q, \sigma, q' \rangle \in \Delta \right\} & \text{if } q \notin F \text{ and } p(q) \neq p \end{cases}$$

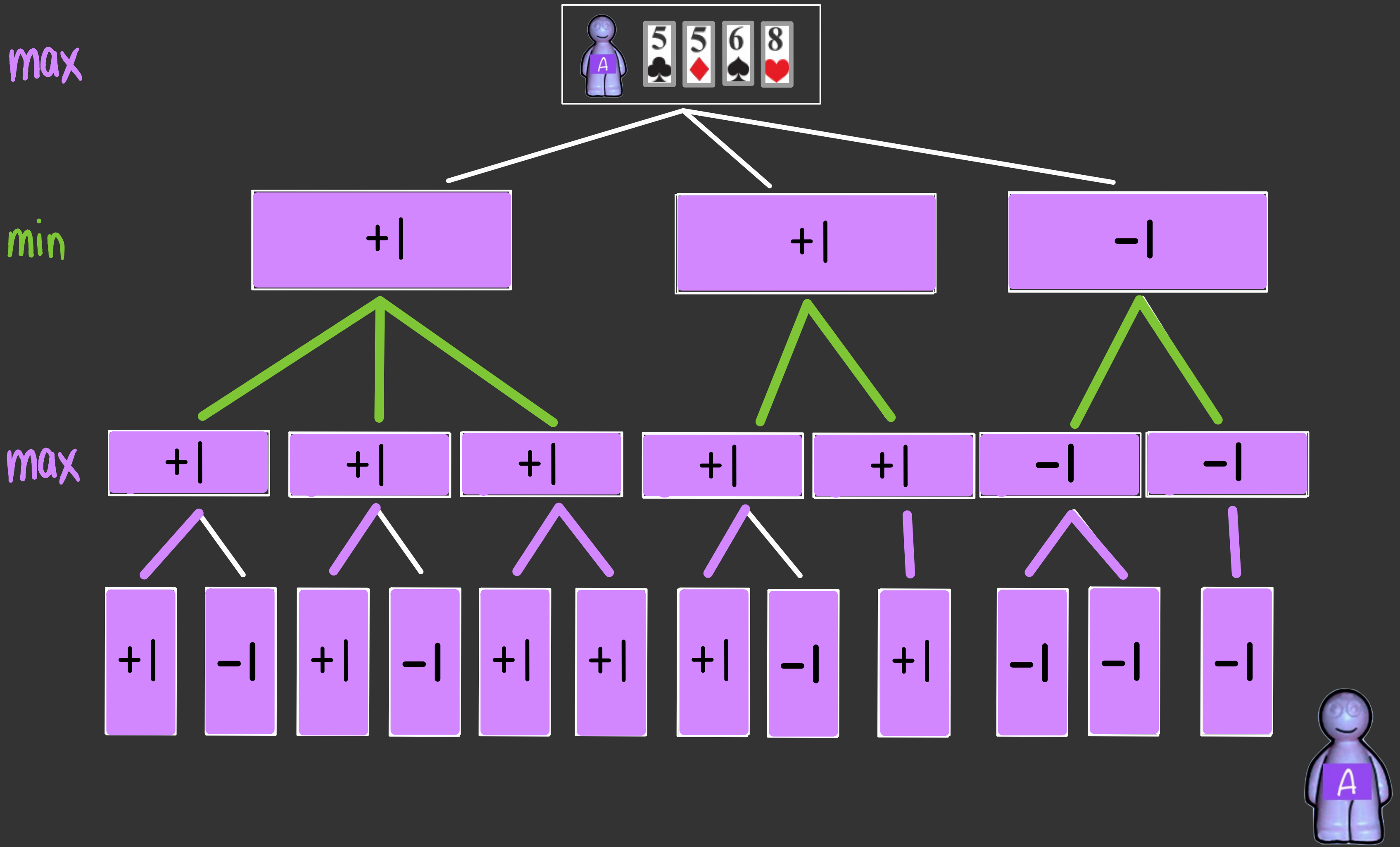


max
min
max
+1
-1
+1
-1
+1
+1
+1
-1
+1
-1
-1
-1

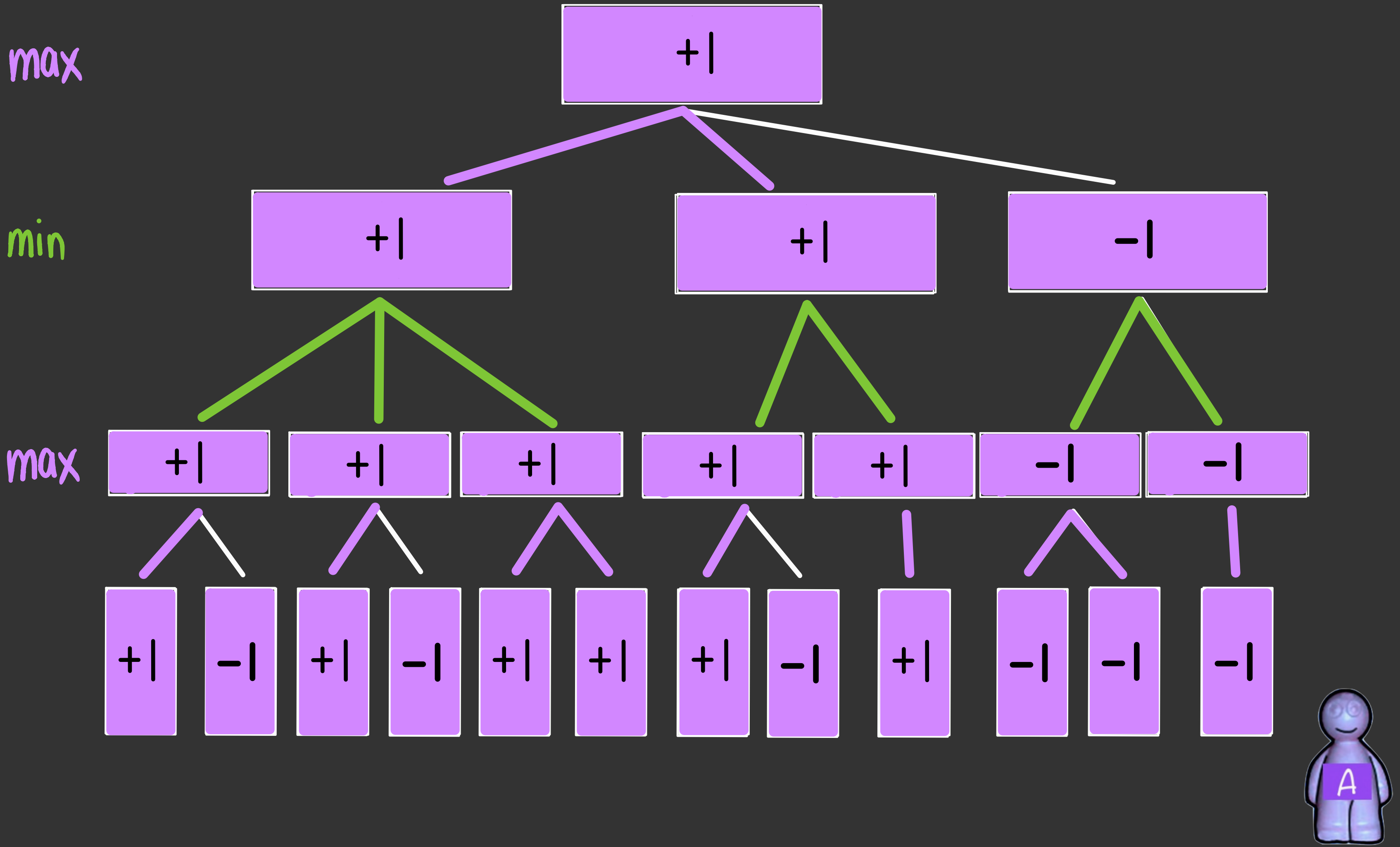

max
min
max
A
B
+1
-1
+1
-1
+1
+1
+1
-1
+1
-1
-1
-1

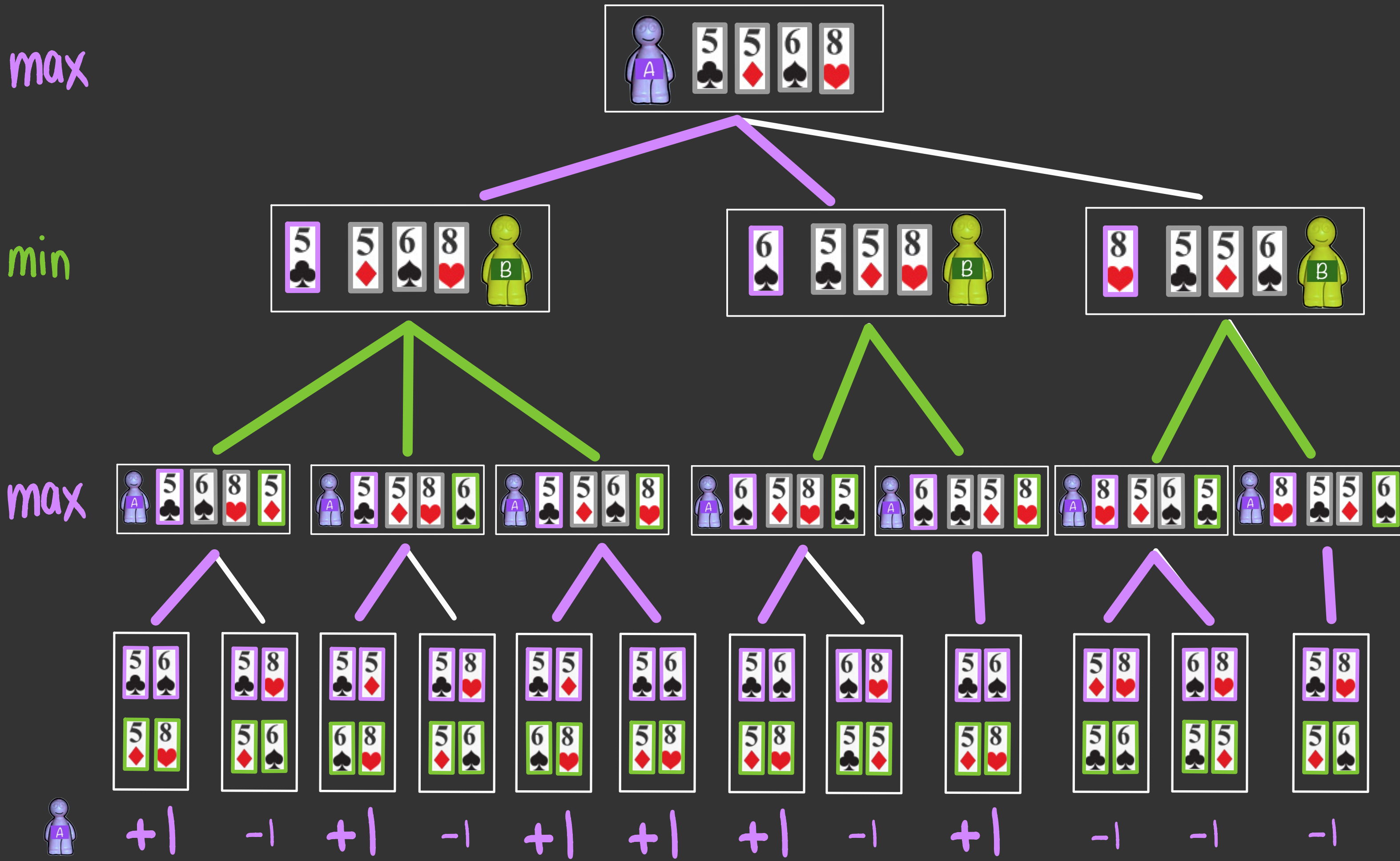


max
min
max
A
5 5 6 8
5 5 6 8 B
6 5 5 8 B
8 5 5 6 B
+1 +1 +1 +1 +1 −1 −1
+1 −1 +1 −1 +1 +1 +1 −1 +1 −1 −1 −1
A



max
min
max
5
5
6
8
A
+1
+1
−1
+1
+1
+1
+1
+1
−1
−1
+1
−1
+1
−1
+1
+1
+1
−1
+1
−1
−1
−1
A


max
+1
min
+1
+1
−1
max
+1
+1
+1
+1
+1
−1
−1
+1
−1
+1
−1
+1
+1
+1
−1
+1
−1
−1
−1
A


max
min
max
A
B
+1
-1
+1
-1
+1
+1
+1
-1
+1
-1
-1
-1

$$
\text{minimax}(q,p) = \begin{cases} U(q,p) & \text{if } q \in F \\ \max\limits_{q'} \left\{ \text{minimax}(q',p) \,\middle|\, \langle q,\sigma,q' \rangle \in \Delta \right\} & \text{if } q \notin F \text{ and } p(q) = p \\ \min\limits_{q'} \left\{ \text{minimax}(q',p) \,\middle|\, \langle q,\sigma,q' \rangle \in \Delta \right\} & \text{if } q \notin F \text{ and } p(q) \neq p \end{cases}
$$

or, as pseudocode:

MINIMAX $(q, p, (m, U))$:

- if $q \in F$: return $U(q,p)$
- children $= \left\{ \text{minimax}(q', p, (m, U)) \,\middle|\, \langle q, \sigma, q' \rangle \in \Delta(m) \right\}$
- return max(children) if $p(q) = p$ else min(children)

$$\text{minimax}(q,p) = \begin{cases} U(q,p) & \text{if } q \in F \\ \max\limits_{q'} \left\{ \text{minimax}(q',p) \,\middle|\, \langle q,\sigma,q' \rangle \in \Delta \right\} & \text{if } q \notin F \text{ and } p(q)=p \\ \min\limits_{q'} \left\{ \text{minimax}(q',p) \,\middle|\, \langle q,\sigma,q' \rangle \in \Delta \right\} & \text{if } q \notin F \text{ and } p(q)\neq p \end{cases}$$

what is this?

or, as pseudocode:

MINIMAX$(q, p, (m, U))$:

- if $q \in F$: return $U(q,p)$
- children = $\{ \text{minimax}(q', p, (m, U)) \mid \langle q, \sigma, q' \rangle \in \Delta(m) \}$
- return max(children) if $p(q) = p$ else min(children)

$$\mathrm{minimax}(q,p) = \begin{cases} U(q,p) & \text{if } q \in F \\ \max_{q'} \{\mathrm{minimax}(q',p) \mid \langle q,\sigma,q'\rangle \in \Delta\} & \text{if } q \notin F \text{ and } p(q) = p \\ \min_{q'} \{\mathrm{minimax}(q',p) \mid \langle q,\sigma,q'\rangle \in \Delta\} & \text{if } q \notin F \text{ and } p(q) \neq p \end{cases}$$

a game with players P is a tuple (m, U) where:

- $m = (Q, \Sigma, \Delta, q_0, F)$ is a state machine where each state $q \in Q$ has the form $(p, q') \in P \times Q'$ for an auxiliary set Q' of states
- utility function $U: F \times P \to \mathbb{R}$ which gives the value of each final state for each player

recall

or, as pseudocode:

MINIMAX$(q, p, (m, U))$:

- if $q \in F$: return $U(q,p)$
- children $= \{\mathrm{minimax}(q', p, (m, U)) \mid \langle q, \sigma, q'\rangle \in \Delta(m)\}$
- return max(children) if $p(q) = p$ else min(children)