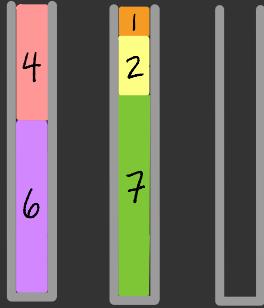


multiplayer  
games

21 sept  
2022

CSCI  
373

train  
brain  
brawn  
prawn



all of the  
tasks we've  
discussed so far  
have involved a  
**single  
agent**

but what if others  
want to stop us from  
succeeding?



"spy vs. spy"  
mad magazine

# the game of nim

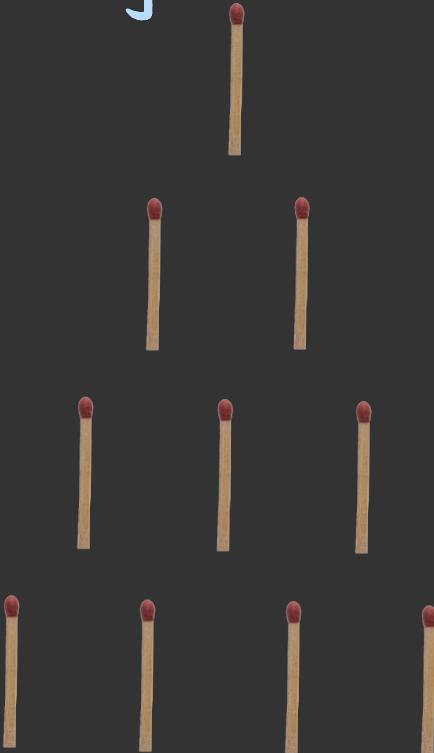
on your turn, you  
must remove one or  
more matchsticks  
from a single row



you lose if you  
remove the last  
matchstick

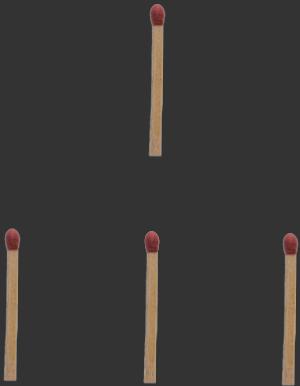
# the game of nim

on your turn, you  
must remove one or  
more matchsticks  
from a single row



you lose if you  
remove the last  
matchstick

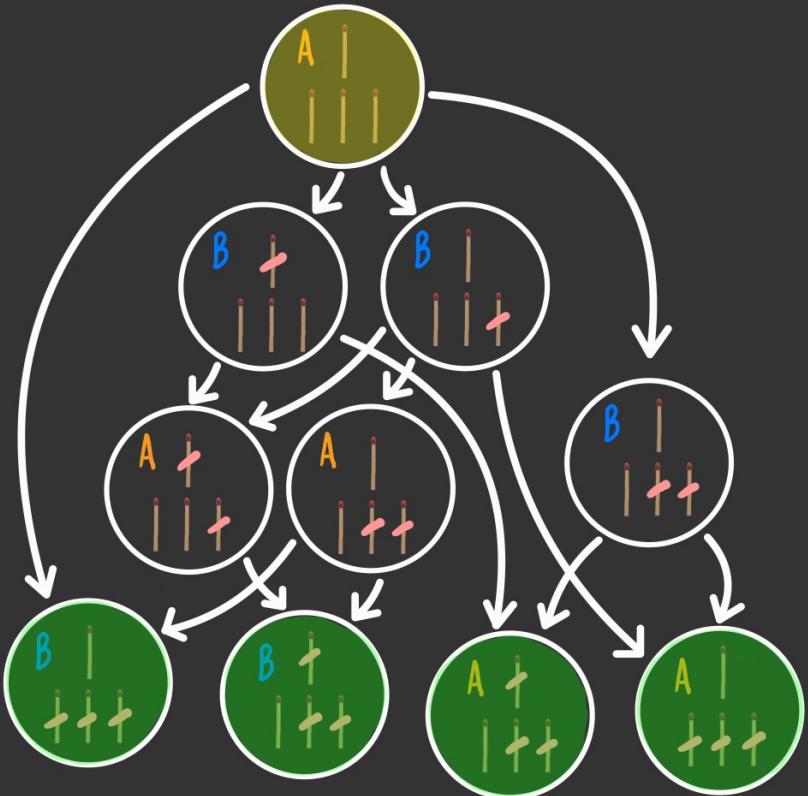
# the game of nim



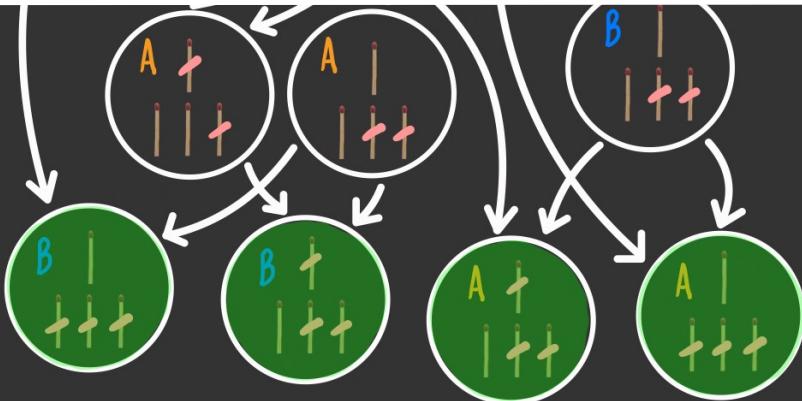
assume this initial nim board

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  $U: F \times P \rightarrow \mathbb{R}$  which gives the value of each final state for each player



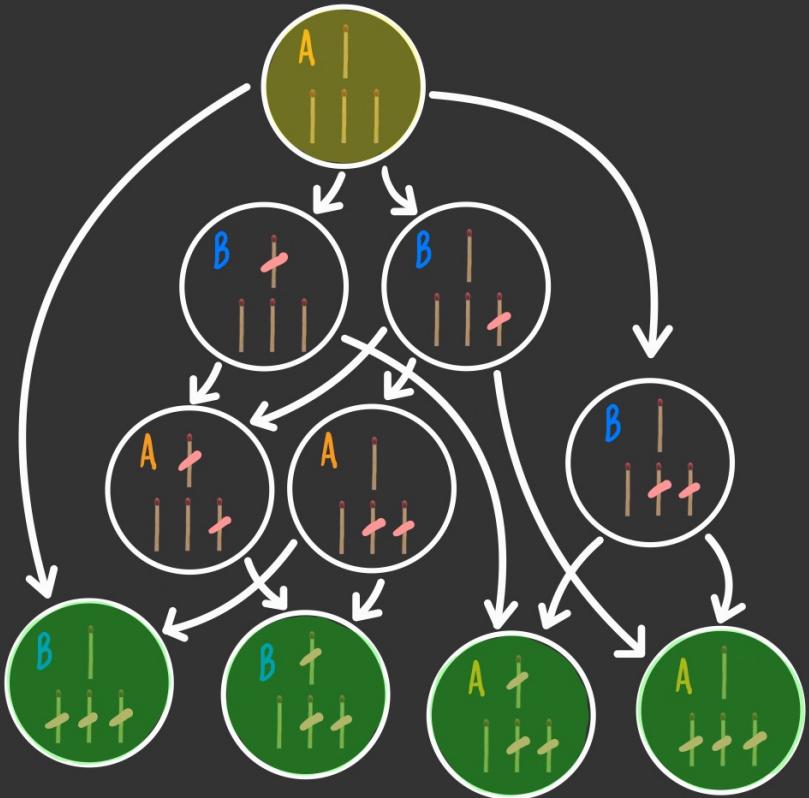
- $S_0$ : The **initial state**, which specifies how the game is set up at the start.
- $\text{TO-MOVE}(s)$ : The player whose turn it is to move in state  $s$ .
- $\text{ACTIONS}(s)$ : The set of legal moves in state  $s$ .
- $\text{RESULT}(s, a)$ : The **transition model**, which defines the state resulting from taking action  $a$  in state  $s$ .
- $\text{IS-TERMINAL}(s)$ : A **terminal test**, which is true when the game is over and false otherwise. States where the game has ended are called **terminal states**.
- $\text{UTILITY}(s, p)$ : A **utility function** (also called an objective function or payoff function), which defines the final numeric value to player  $p$  when the game ends in terminal state  $s$ . In chess, the outcome is a win, loss, or draw, with values 1, 0, or  $1/2$ .<sup>2</sup> Some games have a wider range of possible outcomes—for example, the payoffs in backgammon range from 0 to 192.



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

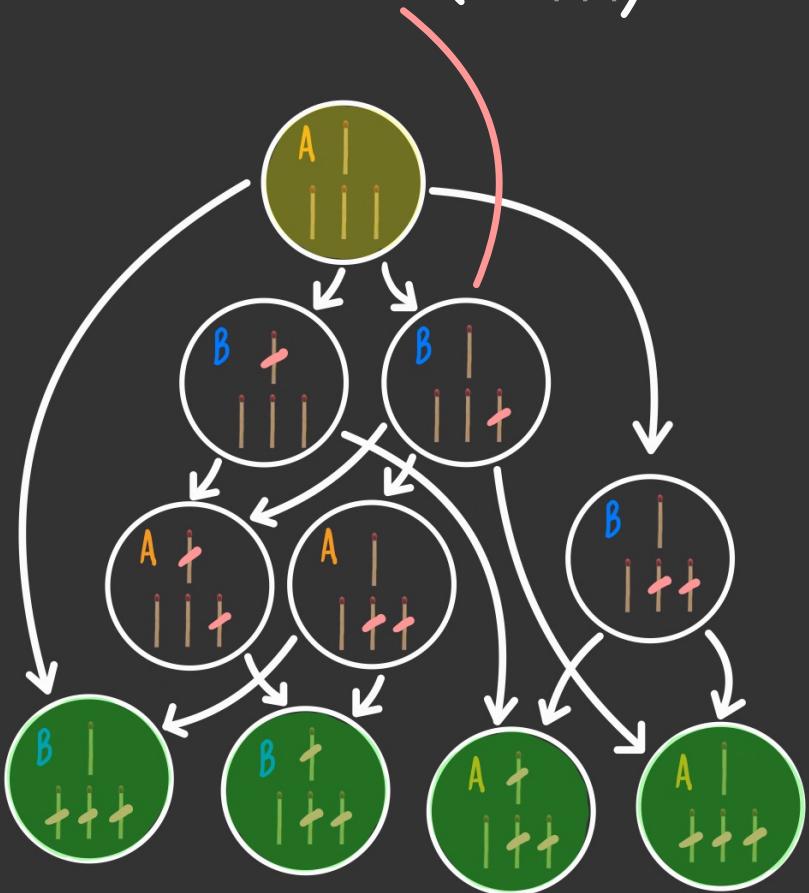
- $M = (Q, \Sigma, \Delta, q_0, F)$  is a state machine where each state  $q \in Q$  has the form  $(p, q') \in P \times Q'$  for an auxiliary set  $Q'$  of states
- utility function  $U: F \times P \rightarrow \mathbb{R}$  which gives the value of each final state for each player

$$P = \{A, B\}$$



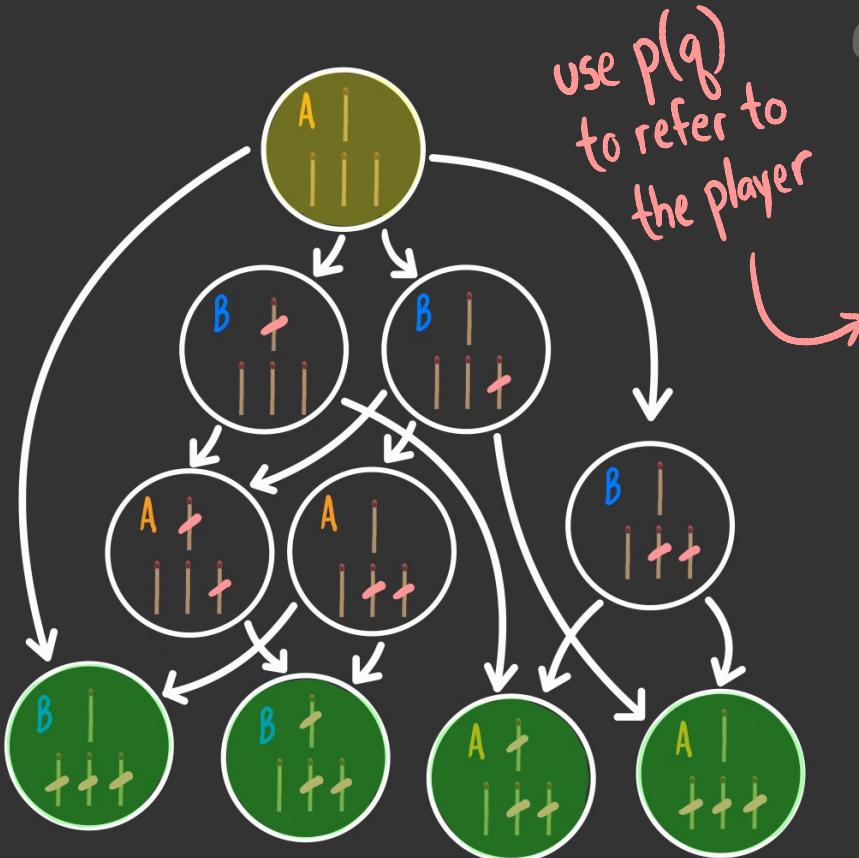
- a game with players  $P$  is a tuple  $(M, U)$  where:
- $M = (Q, \Sigma, \Delta, q_0, F)$  is a state machine where each state  $q \in Q$  has the form  $(p, q') \in P \times Q'$  for an auxiliary set  $Q'$  of states
  - utility function  $U: F \times P \rightarrow \mathbb{R}$  which gives the value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

shorthand for  $(B, \text{ ||+ })$

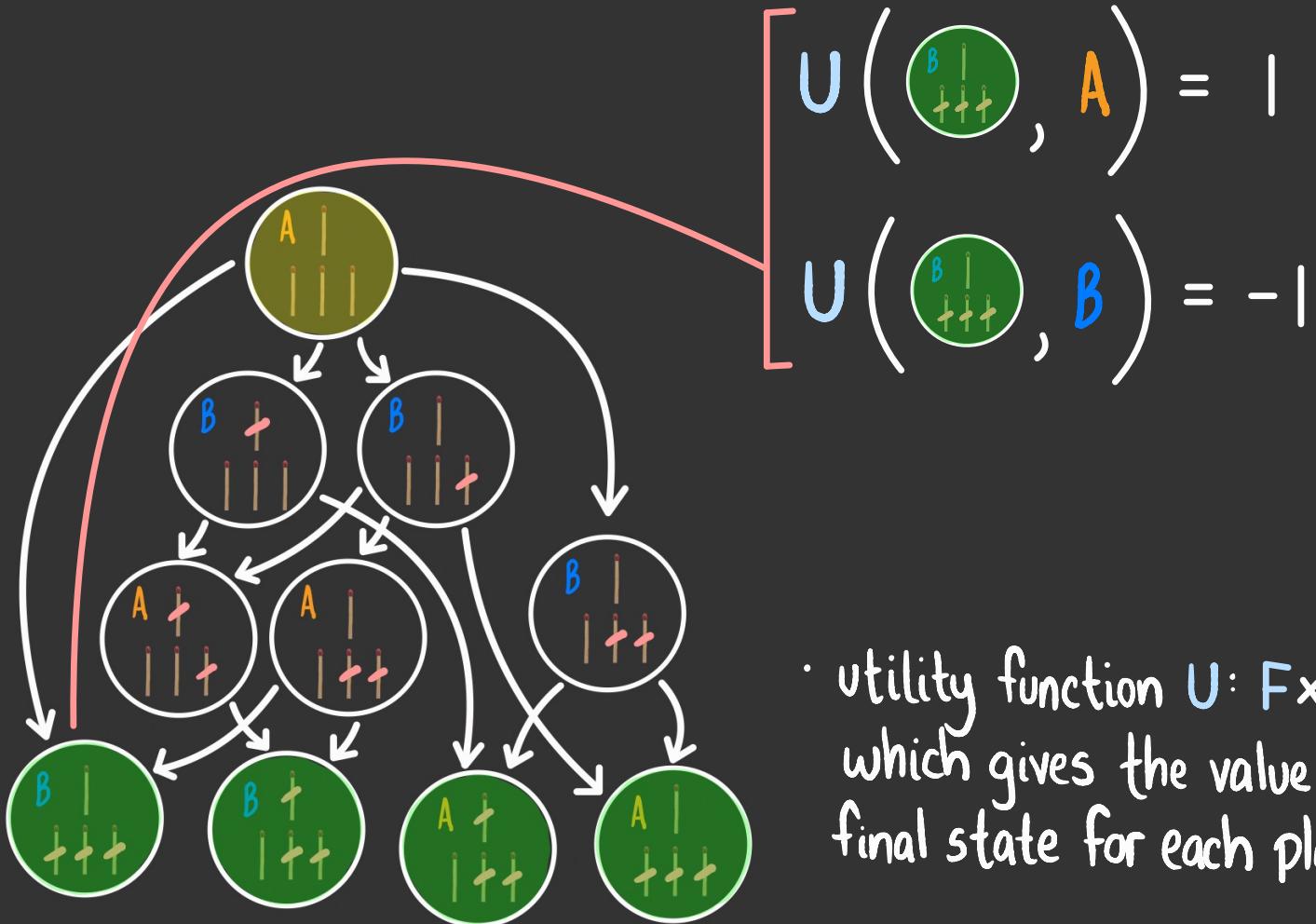


- a game with players  $P$  is a tuple  $(M, U)$  where:
- $M = (Q, \Sigma, \Delta, q_0, F)$  is a state machine where each state  $q \in Q$  has the form  $(p, q') \in P \times Q'$  for an auxiliary set  $Q'$  of states
  - utility function  $U: F \times P \rightarrow \mathbb{R}$  which gives the value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

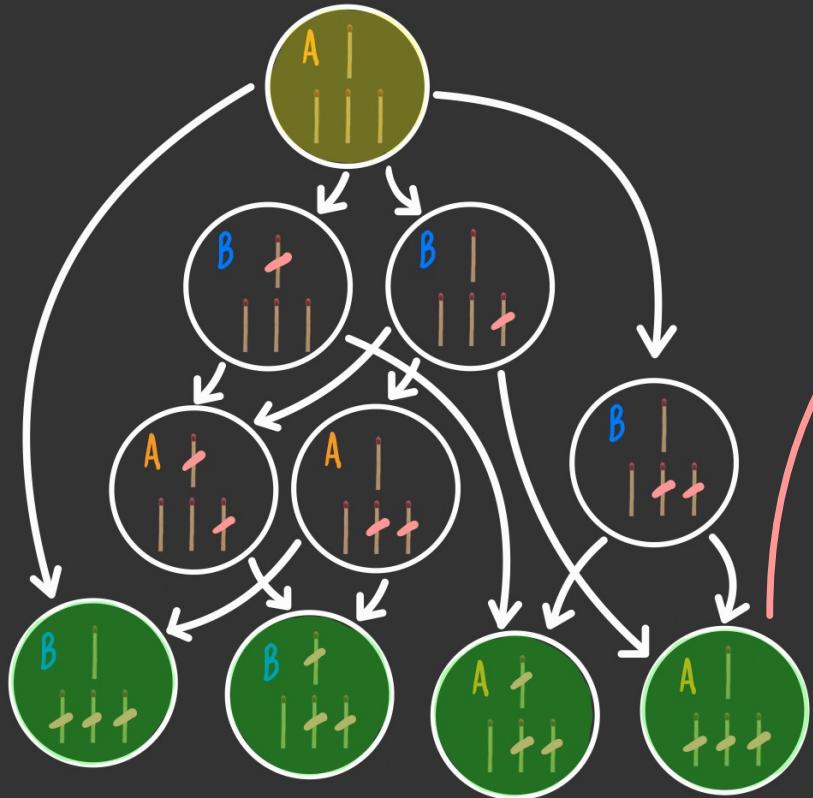
e.g.  $P \left( \begin{array}{c} B \\ | \\ \text{---} \\ | \\ \text{---} \\ | \\ \text{---} \end{array} \right) = B$



- 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  $U: F \times P \rightarrow \mathbb{R}$  which gives the value of each final state for each player



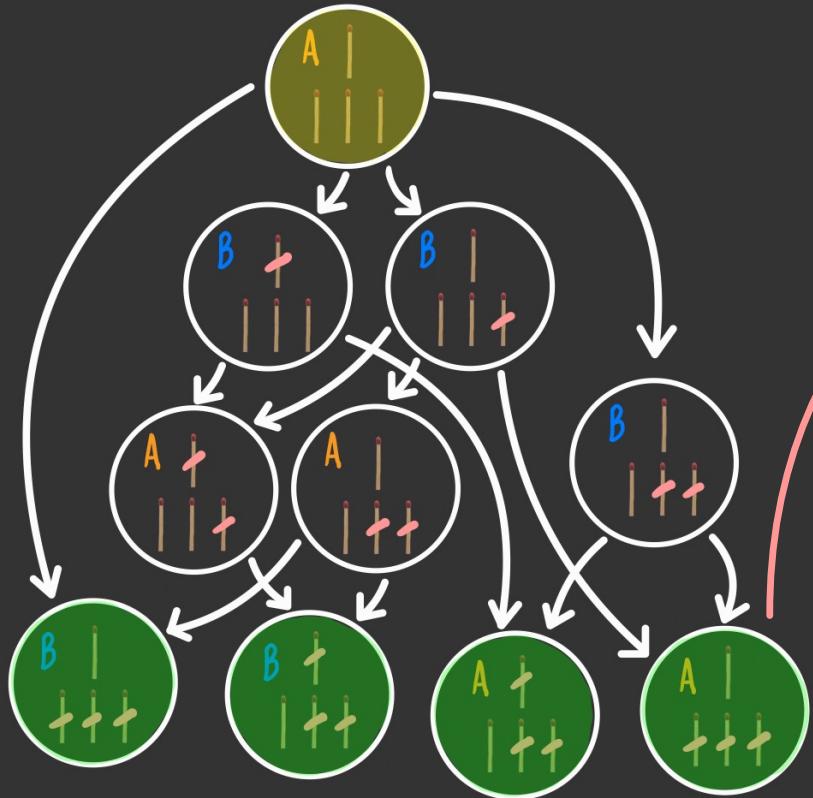
- utility function  $U: F \times P \rightarrow \mathbb{R}$   
which gives the value of each final state for each player



$$U(A, A) = ?$$

$$U(A, B) = ?$$

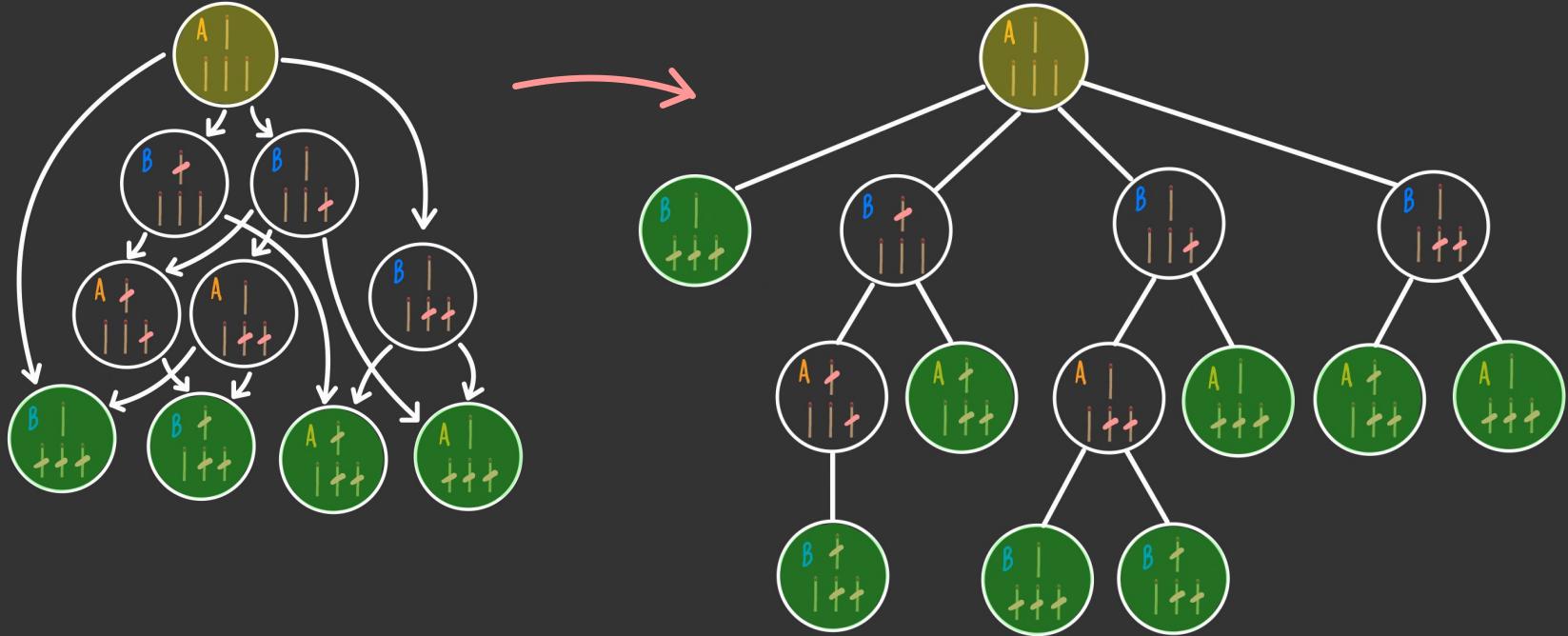
- utility function  $U: F \times P \rightarrow \mathbb{R}$   
which gives the value of each final state for each player



$$U(A, A) = -1$$

$$U(A, B) = 1$$

- utility function  $U: F \times P \rightarrow \mathbb{R}$   
which gives the value of each final state for each player



analogous to single-agent search, a game can be unraveled into a search tree