CMSC 474, Game Theory

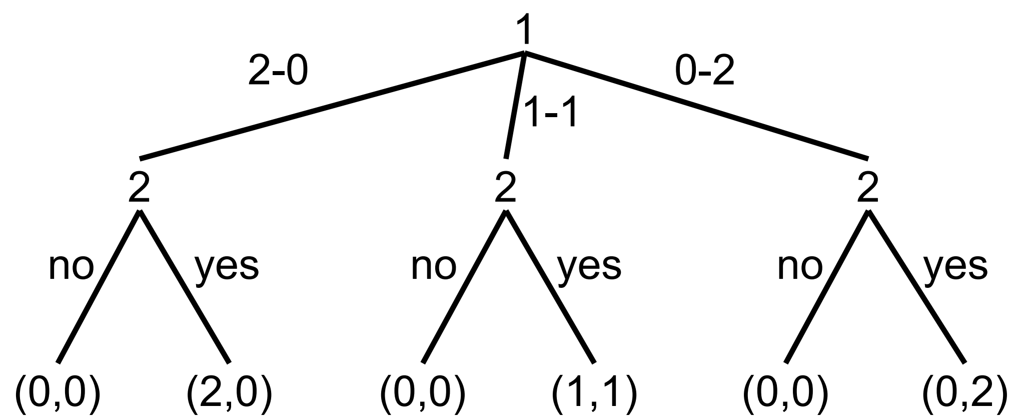
4a. Extensive-Form Games

Dana Nau

University of Maryland

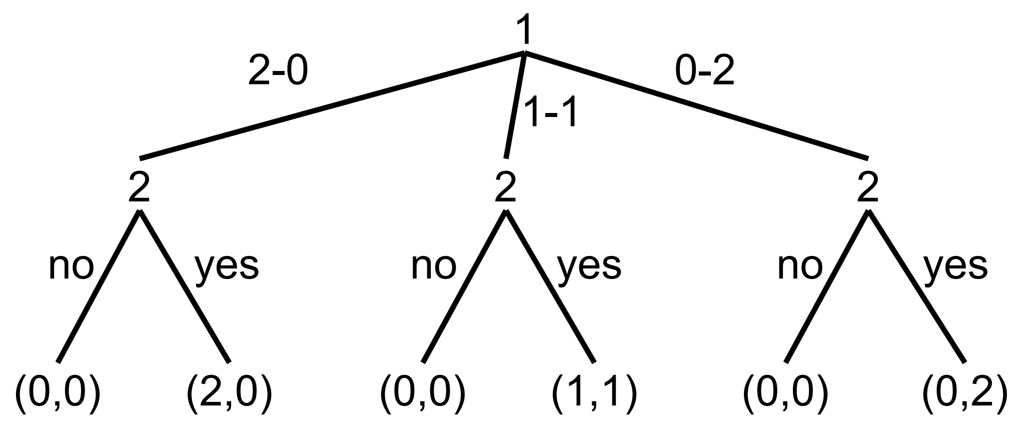
The Sharing Game

- Suppose agents 1 and 2 are two children
- Someone offers them two cookies, but only if they can agree how to share them
- Agent 1 chooses one of the following options:
  - Agent 1 gets 2 cookies, agent 2 gets 0 cookies
  - They each get 1 cookie
  - Agent 1 gets 0 cookies, agent 2 gets 2 cookies
- Agent 2 chooses to accept or reject the split:
  - Accept => they each get their cookies
  - Otherwise, neither gets any



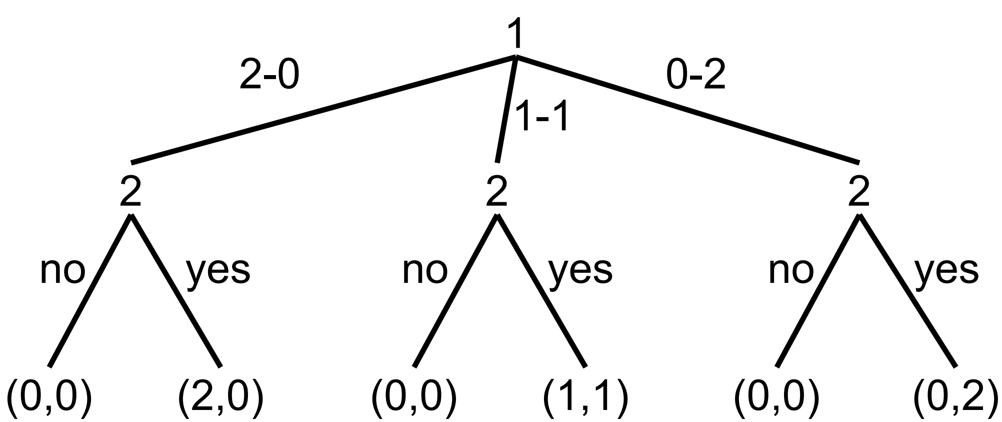
Perfect-Information Extensive Form

- **Extensive form**: make the game's temporal structure explicit
  - Don't assume players choose their strategies all at once
- **Perfect information**:
  - Every agent knows all players' utility functions and possible actions
  - Every agent knows the history and current state
    - no simultaneous actions; agents move one at a time
- Can be converted to normal form
  - So previous results carry over
- But there are additional results that depend on the temporal structure



Perfect-Information Extensive Form

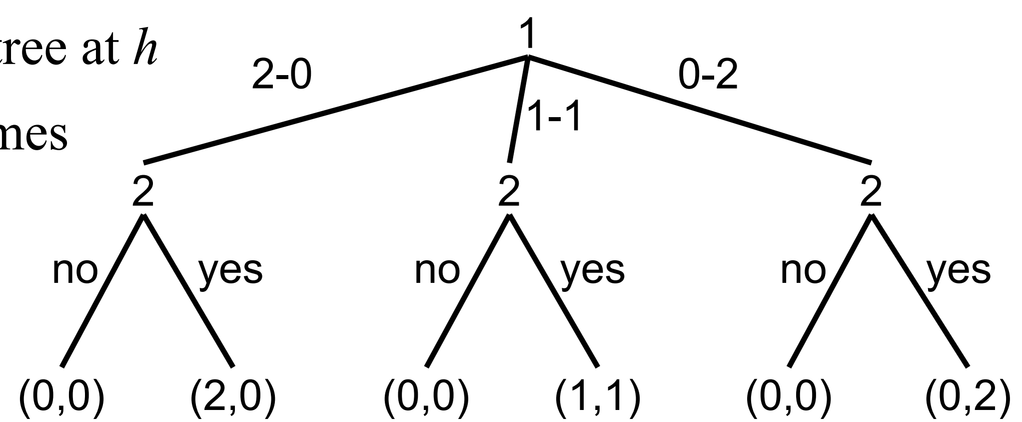
- In a perfect-information game, the extensive form is a **game tree**:
  - **Choice** (or **nonterminal**) **node**: place where an agent chooses an action
    - H = {nonterminal nodes}
  - **Edge**: an available **action** or **move**
  - **Terminal node**: a final outcome
  - At each terminal node h, each agent i has a utility $u_i(h)$



Notation from the Book (Section 4.1)

- H = {nonterminal nodes}
- Z = {terminal nodes}
- If h is a nonterminal node, then
  - $\rho(h)$ = the player to move at h
  - $\chi(h)$ = {all available actions at h}
  - $\sigma(h,a)$ = node produced by action a at node h
  - h's **children** or **successors** = $\{\sigma(h,a) \mid a \in \chi(h)\}$
- If h is a node (either terminal or nonterminal), then
  - h's **history** = sequence of actions from the root to h
  - h's **descendants** = nodes in subtree at h
- The book doesn't give the nodes names
  - The labels tell which agent makes the next move

I'm not used to this notation, might not always remember it

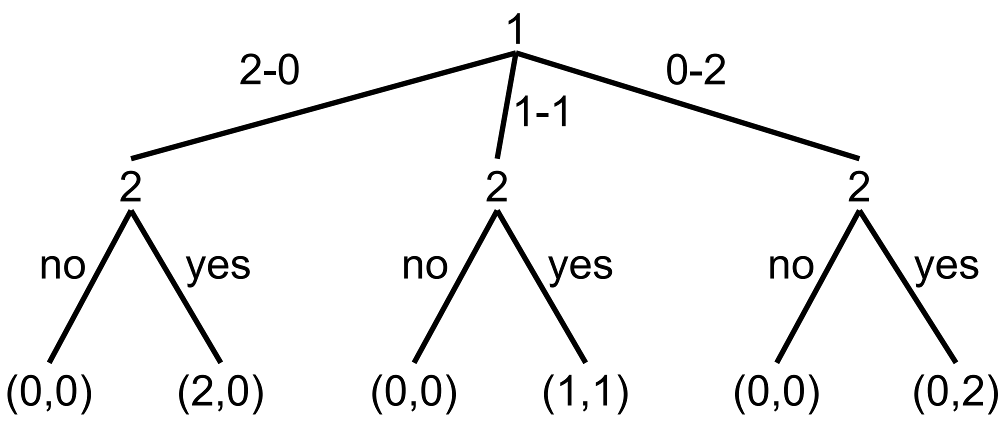


Pure Strategies

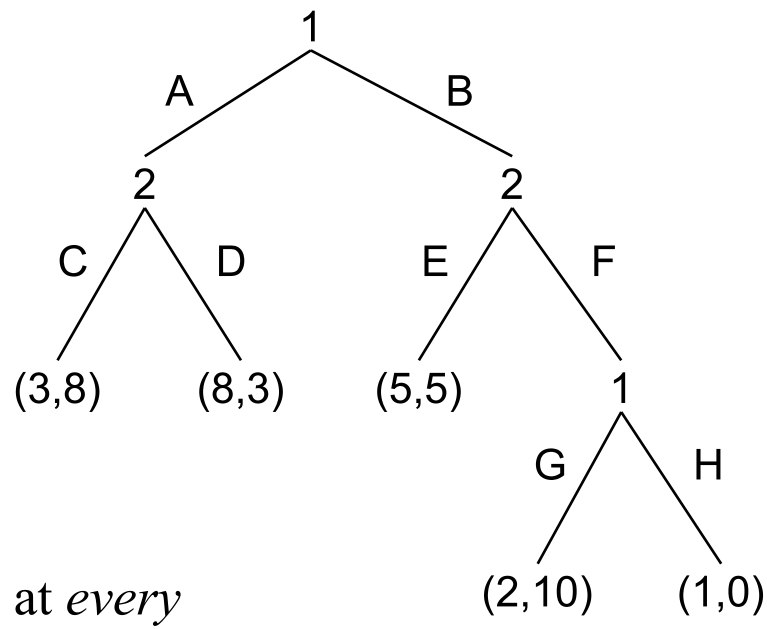
- Pure strategy for agent i in a perfect-information game:
  - Function telling what action to take at **every** node where it's i's choice
    - i.e., every node h at which $\rho(h) = i$

Sharing game:

- Agent 1 has 3 pure strategies: S_1 = {2-0, 1-1, 0-2}
- Agent 2 has 8 pure strategies:
- S_2 = {(yes, yes, yes), (yes, yes, no),
  (yes, no, yes), (yes, no, no),
  (no, yes, yes), (no, yes, no),
  (no, no, yes), (no, no, no)}
- Which action at which node?
  - Either assume a fixed ordering on the nodes, or use different action names at each node



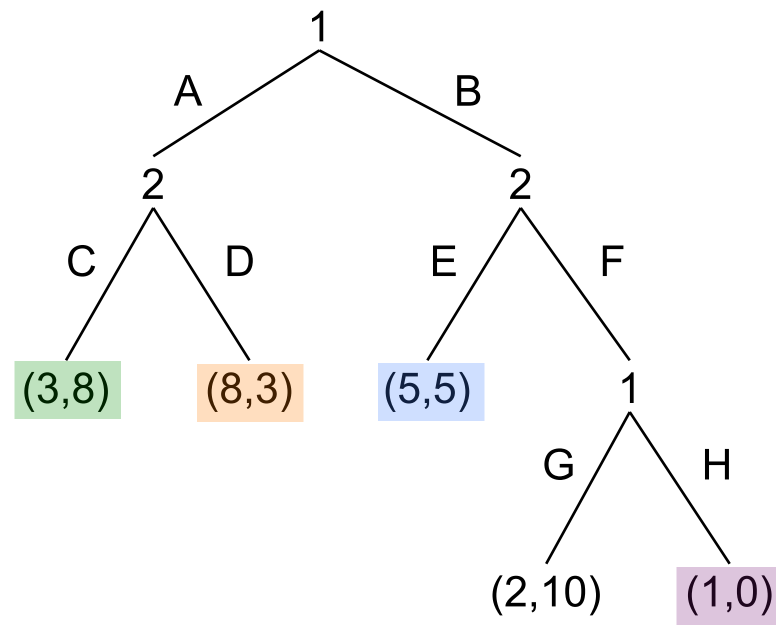
Extensive form vs. normal form



- Every game tree corresponds to an equivalent normal-form game
- To convert
  - Get all of the agents' pure strategies
    - Each strategy must specify an action at *every* node where it's the agent's move
- Example:
  - Agent 1's pure strategies:
    - S_1 = {(A,G), (A,H), (B,G), (B,H)}
      - (A,G) and (A,H) aren't the same strategy
  - Agent 2's pure strategies:
    - S_2 = {(C,E), (C,F), (D,E), (D,F)}

Extensive form vs. normal form

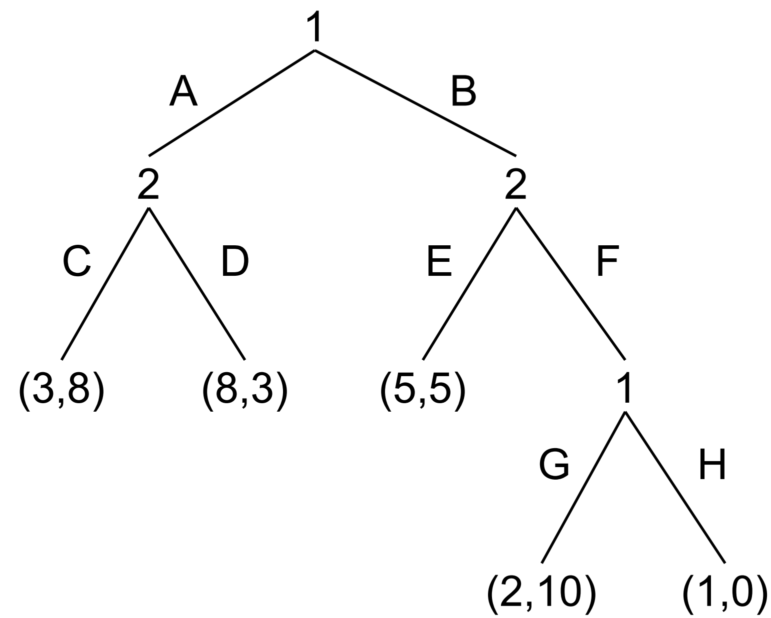
- Next,write the payoff matrix
  - For each strategy profile, see what terminal node it goes to
- Each terminal node may occur several times in the payoff matrix
  - Can cause exponential blowup
    - 5 outcomes in the game tree
    - 16 in the payoff matrix
- 3 pure-strategy Nash equilibria:
  - ((A,G), (C,F))
  - ((A,H), (C,F))
  - ((B,H), (C,E))



	(C,E)	(C,F)	(D,E)	(D,F)
(A,G)	3,8	$\boxed{3,8}$	8,3	8,3
(A,H)	3,8	$\boxed{3,8}$	8,3	8,3
(B,G)	5,5	2,10	5,5	2,10
(B,H)	$\boxed{5,5}$	1,0	5,5	1,0

Nash Equilibrium

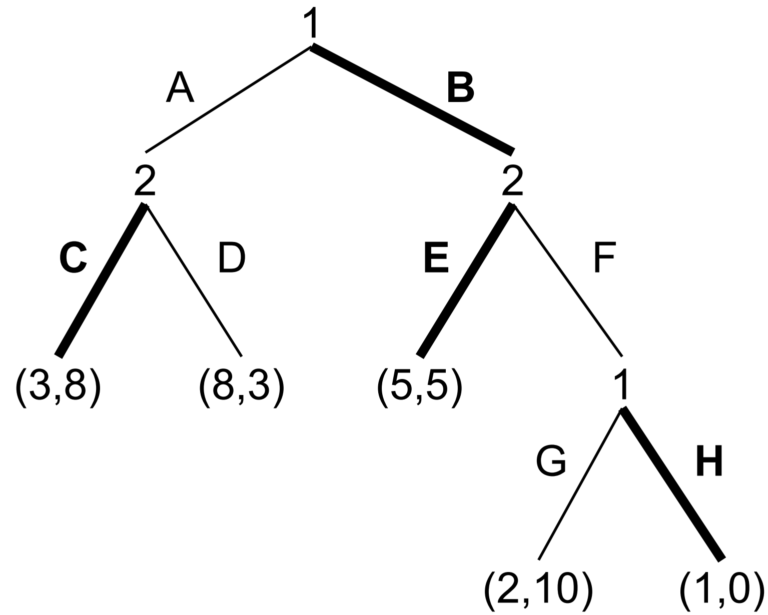
- **Theorem**: Every perfect-information game in extensive form has a *pure-strategy* Nash equilibrium
- Intuition:
  - In mixed-strategy equilibria, the purpose of the mixed strategy
    - Keep the other agents from knowing your action before they choose theirs
  - Not useful in perfect-information games
    - Agents move one at a time
    - Know all previous moves
    - Don't know any subsequent moves



	(C,E)	(C,F)	(D,E)	(D,F)
(A,G)	3,8	$\boxed{3,8}$	8,3	8,3
(A,H)	3,8	$\boxed{3,8}$	8,3	8,3
(B,G)	5,5	2,10	5,5	2,10
(B,H)	$\boxed{5,5}$	1,0	5,5	1,0

Nash Equilibrium

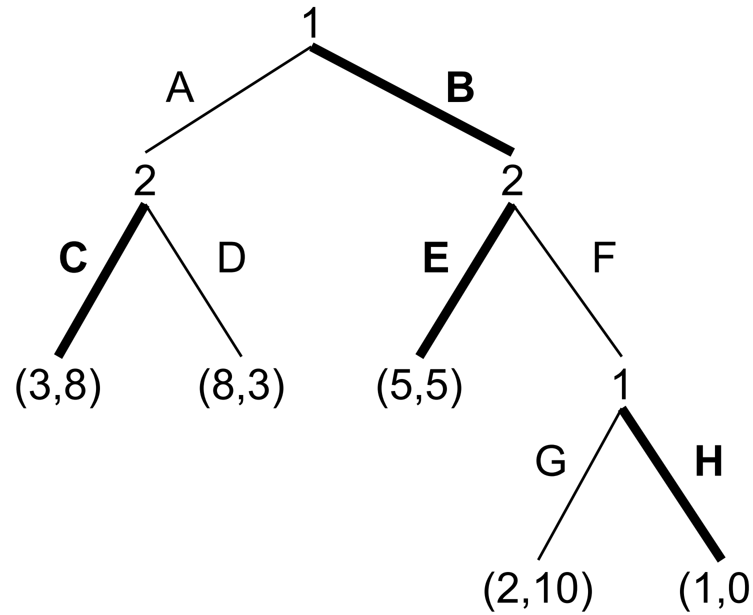
- One of the Nash equilibria is ((B,H), (C,E))
- **Poll 4.1**: when 1 plays B, what should 2 choose?



	(C,E)	(C,F)	(D,E)	(D,F)
(A,G)	3,8	$\boxed{3,8}$	8,3	8,3
(A,H)	3,8	$\boxed{3,8}$	8,3	8,3
(B,G)	5,5	2,10	5,5	2,10
(B,H)	$\boxed{5,5}$	1,0	5,5	1,0

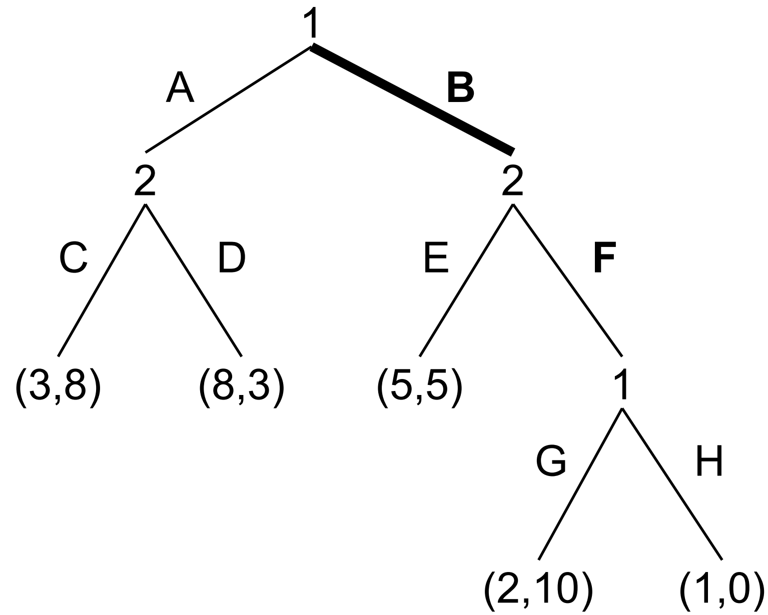
Nash Equilibrium

- One of the Nash equilibria is ((B,H), (C,E))
- If 1's strategy were (B,G)
  - Agent 2's best response would be (C,F)
- When 1 plays B
  - The only reason for 2 to choose E is to keep 1 from doing H
- Suppose that at the start of the game, 1 announces that his/her strategy is (B,H)
  - Agent 1 is making a **threat** that Agent 2 will get 0
  - If 2 believes the threat, 2 will avoid that part of the tree
  - Agent 1 gets 5 instead of ≤ 2



	(C,E)	(C,F)	(D,E)	(D,F)
(A,G)	3,8	3,8	8,3	8,3
(A,H)	3,8	3,8	8,3	8,3
(B,G)	5,5	2,10	5,5	2,10
(B,H)	5,5	1,0	5,5	1,0

Nash Equilibrium

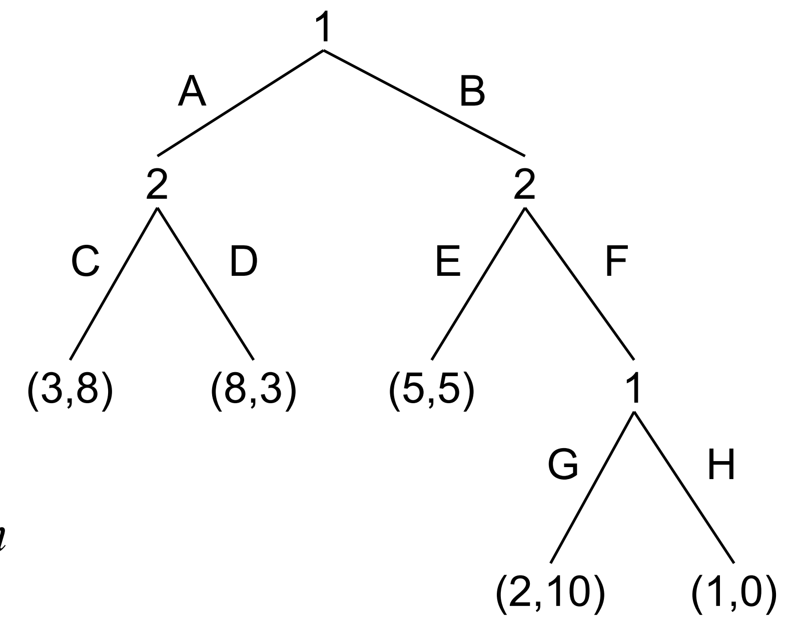


- Is the threat credible?
- If 1 plays B and 2 plays F
  - Will 1 *really* play H rather than G?
  - Not rational: it would reduce 1's utility

	(C,E)	(C,F)	(D,E)	(D,F)
(A,G)	3,8	$\boxed{3,8}$	8,3	8,3
(A,H)	3,8	$\boxed{3,8}$	8,3	8,3
(B,G)	5,5	2,10	5,5	2,10
(B,H)	$\boxed{5,5}$	1,0	5,5	1,0

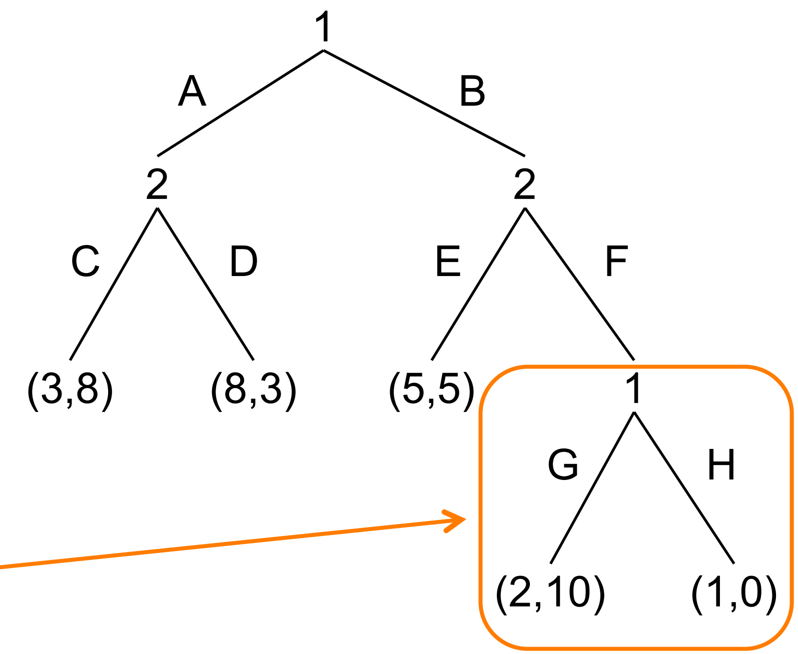
- Need a new solution concept
  - Modified version of Nash equilibrium
  - Exclude non-credible threats

Subgame-Perfect Equilibrium



- Let G be a perfect-information extensive-form game
- **Subgame** of G at node h:
  - restriction of G to the subtree rooted at h
- **Subgame-perfect equilibrium (SPE)**:
  - Strategy profile **s** such that for every subgame of G the restriction of **s** to the subgame is a Nash equilibrium
- No non-credible threats
  - In every subgame, no agent can do better by changing strategy
- Every perfect-information extensive-form game has at least 1 SPE
  - Proof: induction on the height of the game tree

Example

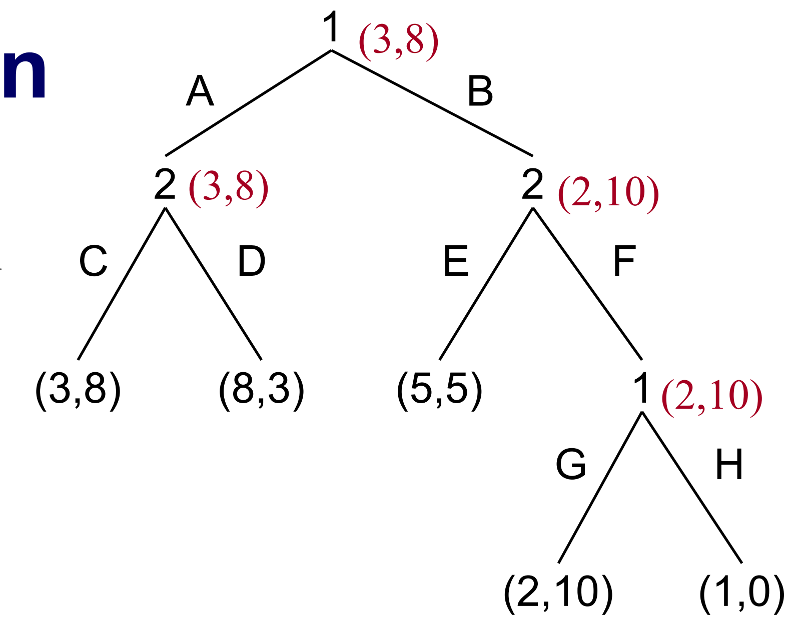


- Recall that we have three Nash equilibria:
  ((A, G), (C, F))
  ((A, H), (C, F))
  ((B, H), (C, E))
- Consider this subgame:
  - H can't be part of a Nash equilibrium
- Excludes ((A,H), (C,F)) and ((B,H), (C,E))
- Just one subgame-perfect equilibrium
  - ((A,G), (C,F))

- To find subgame-perfect equilibria, use **backward induction**

	(C,E)	(C,F)	(D,E)	(D,F)
(A,G)	3,8	$\boxed{3,8}$	8,3	8,3
(A,H)	3,8	~~3,8~~	8,3	8,3
(B,G)	5,5	2,10	5,5	2,10
(B,H)	~~5,5~~	1,0	5,5	1,0

Backward Induction



- At each non-leaf node h:
  - Recursive call to get SPEs for h's children
  - Let h^* = child with highest SPE payoff for player to move at h
  - SPE action at h is to move to h^*

function Backward_Induction(h)
    if $h \in Z$ then return $\mathbf{u}(h)$
    $\mathbf{v}^* \leftarrow [-\infty, -\infty, \ldots, -\infty]$
    for every $a \in \chi(h)$
        $\mathbf{v} \leftarrow$ Backward_Induction($\sigma(h,a)$)
        if $\mathbf{u}[\rho(h)] > \mathbf{v}^*[\rho(h)]$ then $\mathbf{v}^* \leftarrow \mathbf{v}$
    return $\mathbf{v}^*$

- Returns the SPE's payoff profile
  - Can easily modify to get the actions

H = {nonterminal nodes}

Z = {terminal nodes}

$\rho(h)$ = the player to move at node h

$\chi(h)$ = {all available actions at node h}

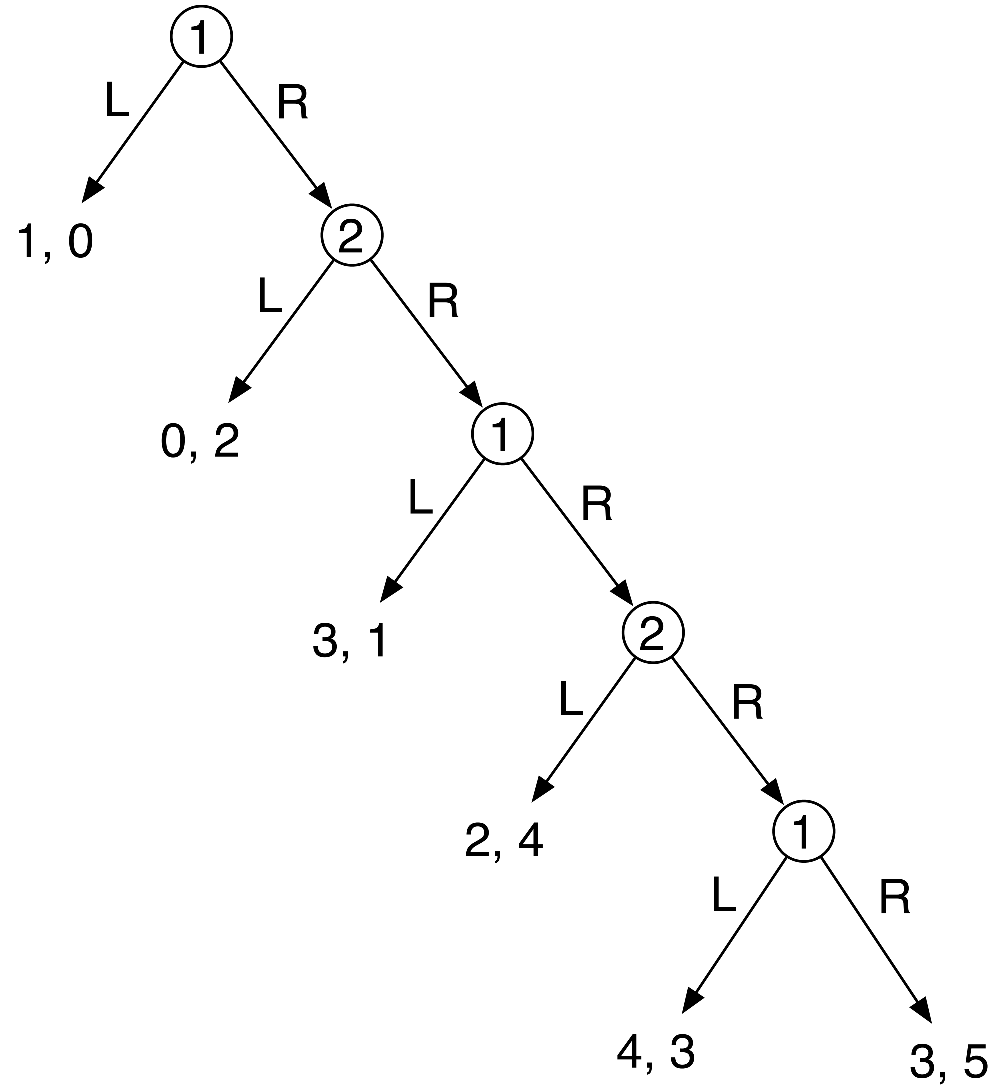
$\sigma(h,a)$ = child of h produced by action a

$\mathbf{u}(h)$ = utility profile at node h

$\mathbf{v}[i]$ = i'th element of utility profile $\mathbf{v}$

The Centipede Game

- I need two volunteers to play a game
- At each nonterminal node, the number tells whose move it is
  - L means *Left*
  - R means *Right*
- At each terminal node, the numbers are your payoffs

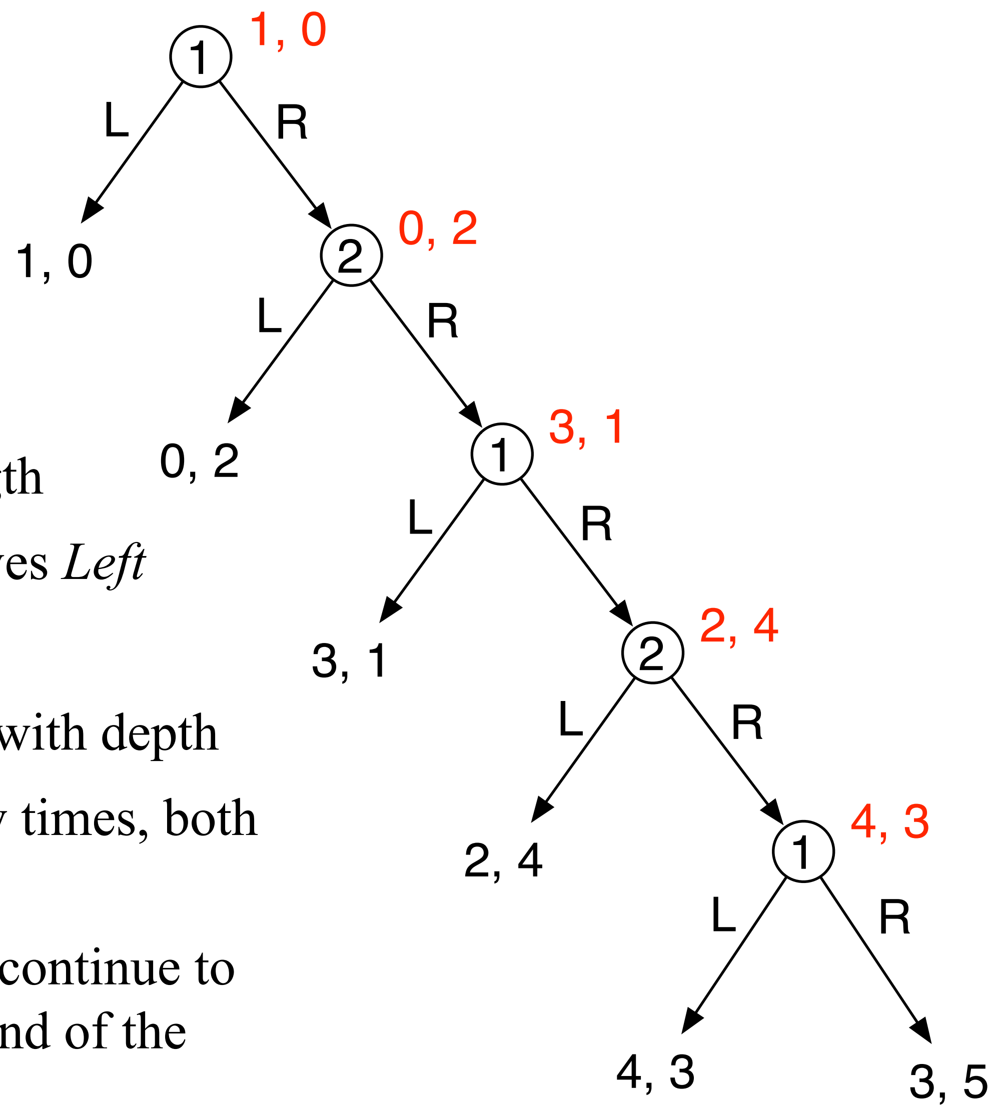


The Centipede Game

- Use backward induction to get the SPE payoffs
- Each player's SPE strategy:
  - Always move *Left*

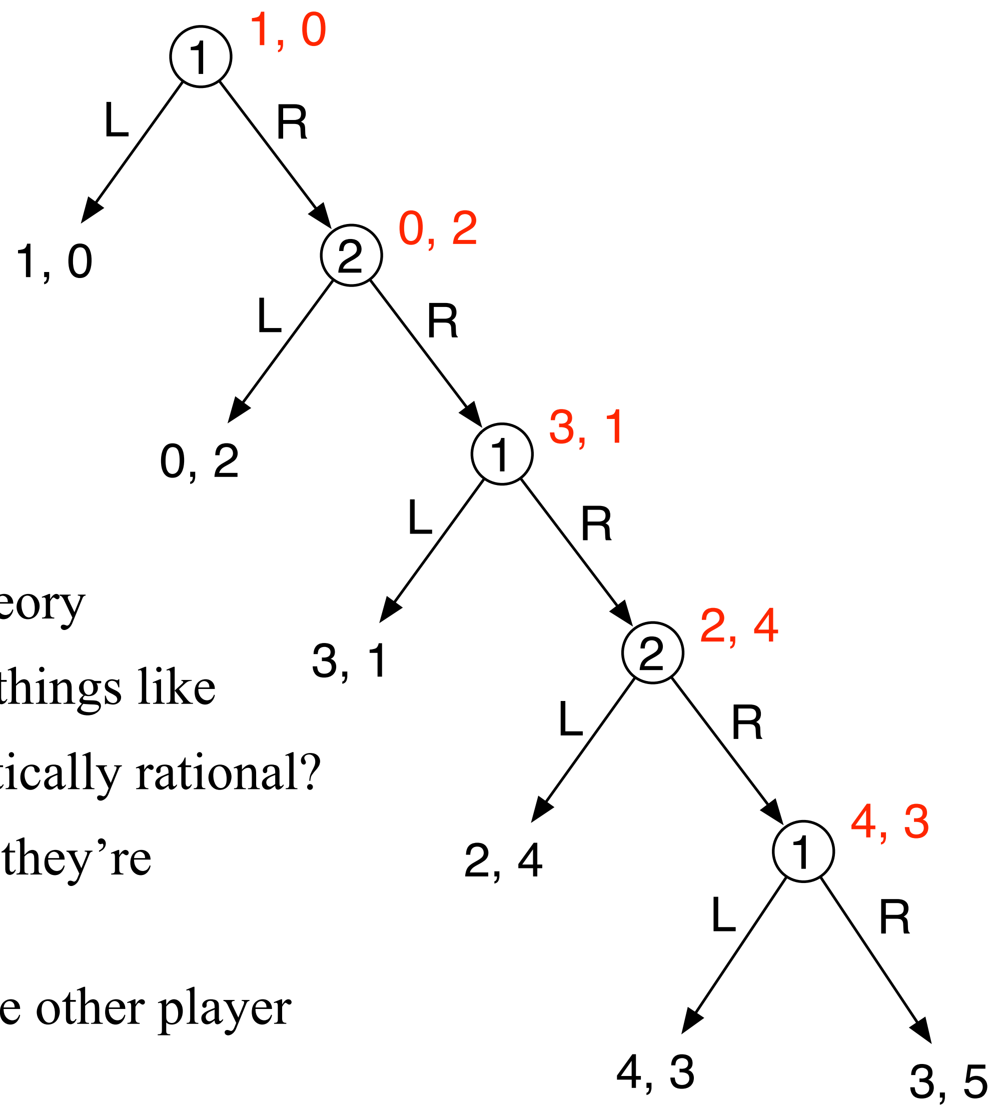
- Can extend the game to any length
  - SPE: each agent always moves *Left*

- $u_1 + u_2$ increases monotonically with depth
  - If both agents go *Right* a few times, both get higher payoffs
  - In lab experiments, subjects continue to choose *Right* until near the end of the game



The Centipede Game

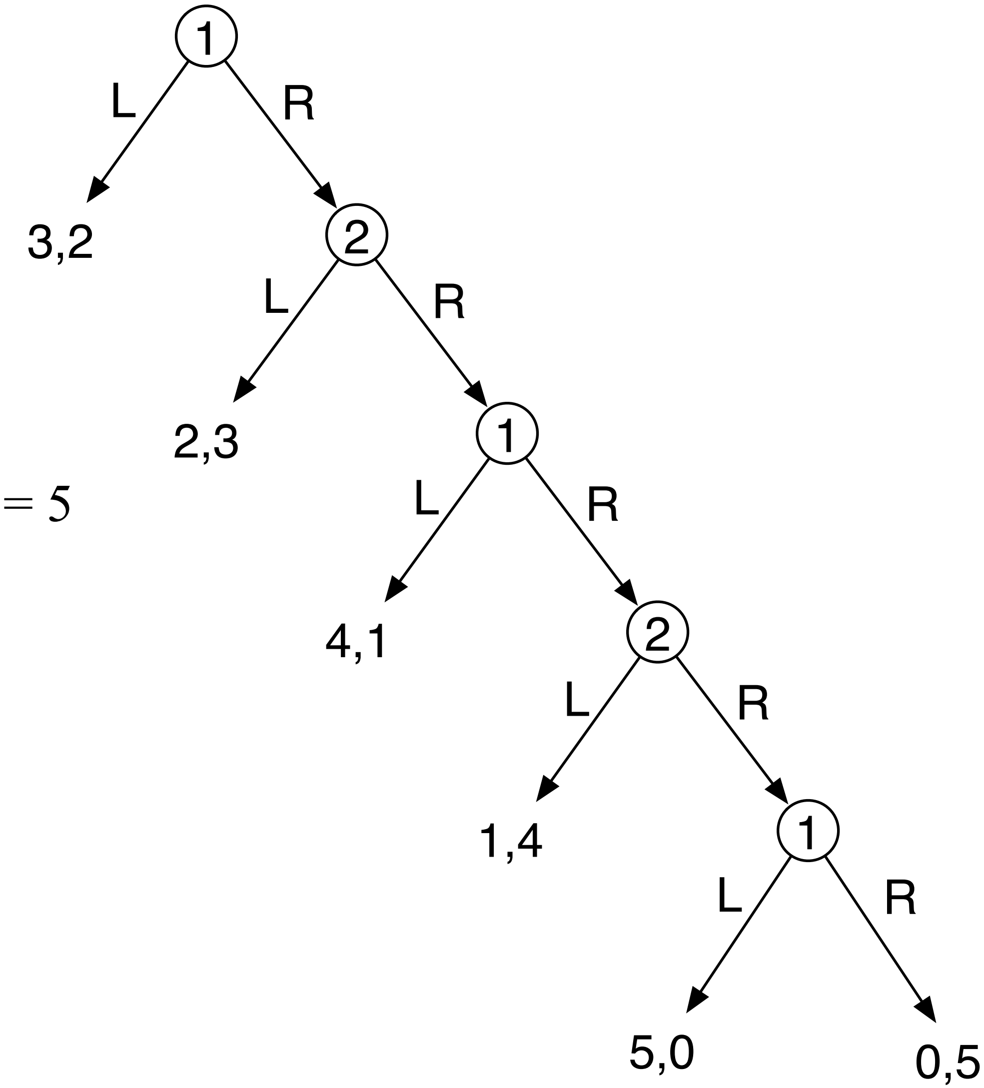
- Suppose agent 1 moves *Right*
- What should agent 2 do?
    - SPE analysis says to move *Left*
    - But it also says we should never be here at all
- Fundamental problem in game theory
- Different answers, depending on things like
    - are both players game-theoretically rational?
    - is it common knowledge that they're game-theoretically rational?
    - how to revise beliefs about the other player from observed behavior



Constant-Sum Centipede Game

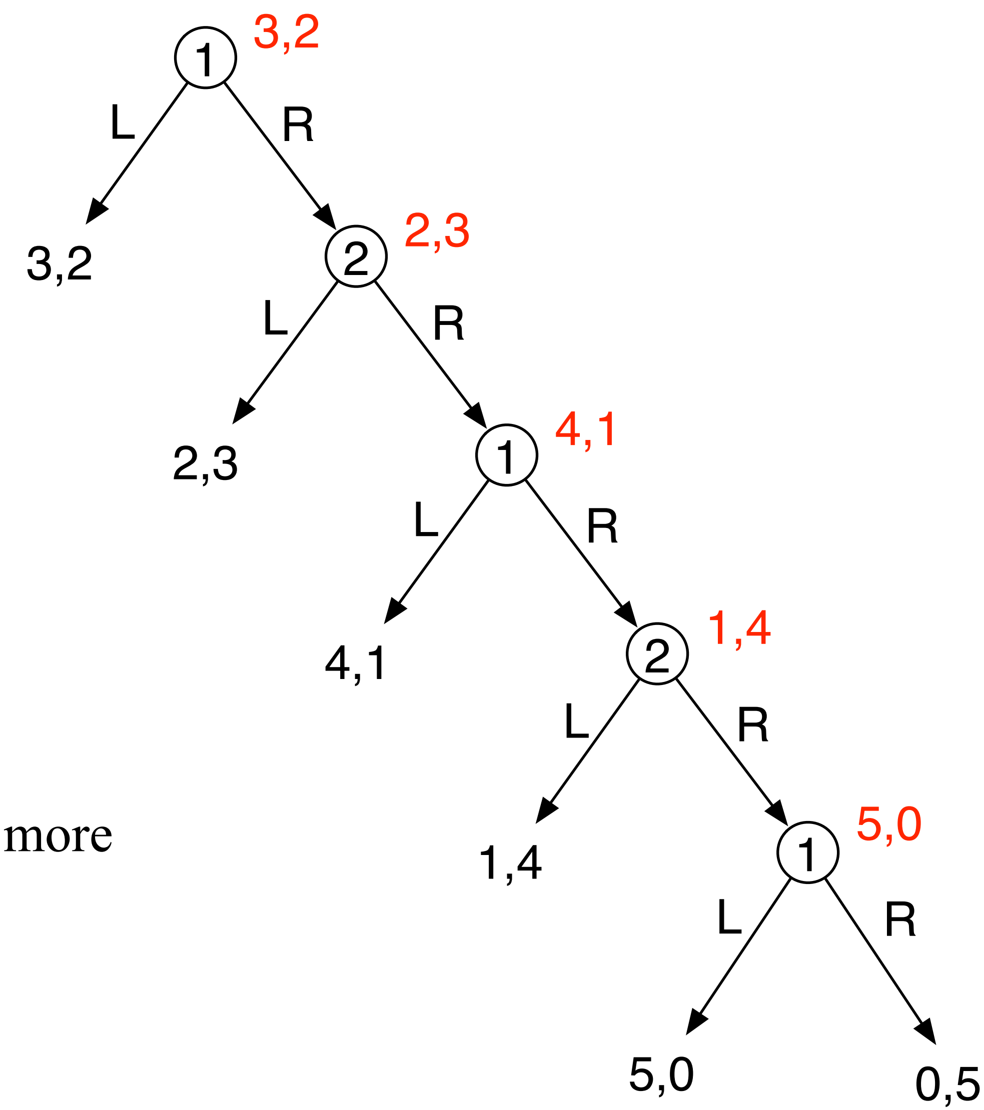
- I need two more volunteers

- At every terminal node, $u_1 + u_2 = 5$



Constant-Sum Centipede Game

- Use backward induction to get the SPE payoffs
- Each player's SPE strategy:
  - Always move *Left*
- Can extend to any depth
  - At every node, $u_1 + u_2 = c$, where $c \geq$ depth of tree
- In this case, SPE strategy gives more accurate results



Minimax Algorithm

- Backward induction is simpler in constant-sum games
  - Only compute u_1
    - $u_2 = -u_1$

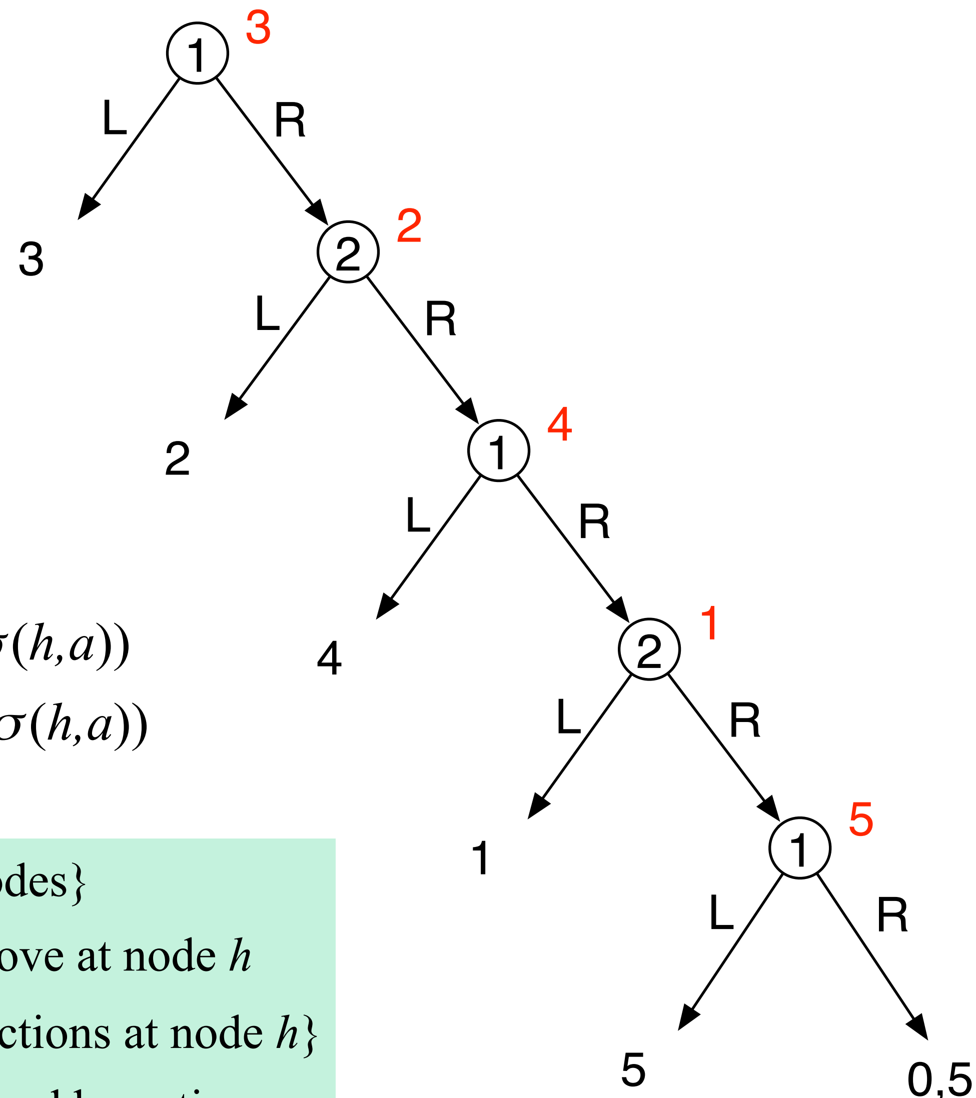
function Minimax(h)

  if $h \in Z$ then return $u(h)$

  else if $\rho(h) = 1$ then

    return $\max_{a \in \chi(h)}$ Minimax($\sigma(h,a)$)

  else return $\min_{a \in \chi(h)}$ Minimax($\sigma(h,a)$)



Z = {terminal nodes}

$\rho(h)$ = player to move at node h

$\chi(h)$ = {available actions at node h}

$\sigma(h,a)$ = node produced by action a

Summary

- Extensive-form games
  - relation to normal-form games
  - Nash equilibria
  - subgame-perfect equilibria
  - backward induction
    - The Centipede Game
  - backward induction in constant-sum games
    - minimax algorithm

- In extensive-form games, the game tree is often too big to search completely
  - E.g., game tree for chess: about 10^{150} nodes
- Lecture 4b (not in book): ways to avoid searching most of the tree