

Alpha/Beta Game Tree Search

Frank Pfenning

15-150, April 14, 2020

Learning Objectives

- Resource limitations
 - Cut off search using value estimators

Learning Objectives

- Resource limitations
 - Cut off search using value estimators
- Algorithmic or implementation improvements exploiting problem-specific properties
 - Alpha/beta pruning

- Bounding search with estimators
- Review minimax search
- Alpha/beta pruning

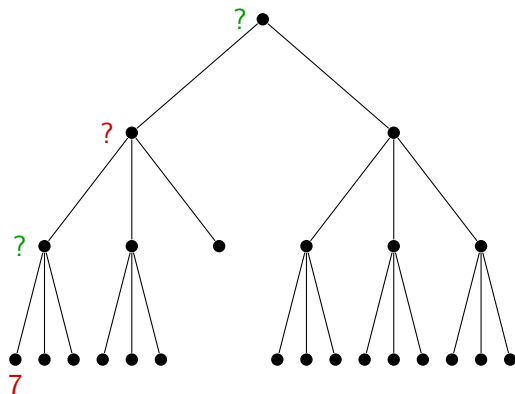
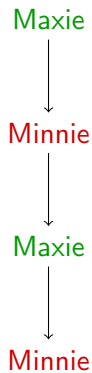
Classes of Games

- 2-player, alternating turns
- Deterministic (no dice)
- Perfect information (no hidden state)
- Zero-sum (A wins iff B loses, or tie)
- Finitely branching
- Examples: tic-tac-toe, connect4, checkers, chess, go, ...

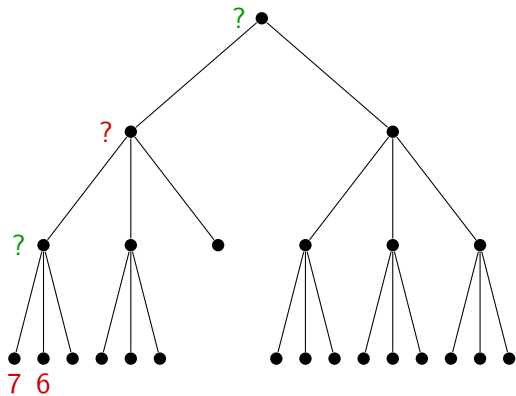
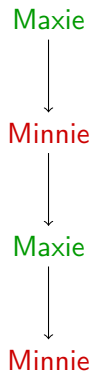
- In practice, we cannot explore the full tree for interesting games
- We cut off exploration (based on various criteria) and **estimate** the value of the position
- Propagate the value up the tree
- Better estimators (generally) result in better players

- In practice, we cannot explore the full tree for interesting games
- We cut off exploration (based on various criteria) and **estimate** the value of the position
- Propagate the value up the tree
- Better estimators (generally) result in better players
- Let's review some code . . .

Minimax Search



Minimax Search



Minimax Search

Maxie



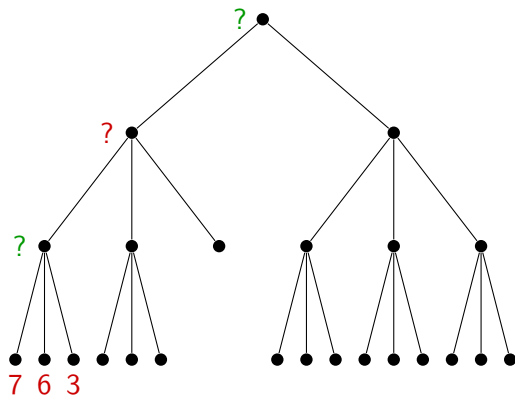
Minnie



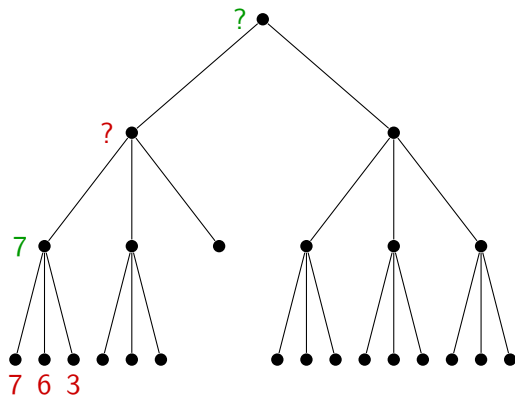
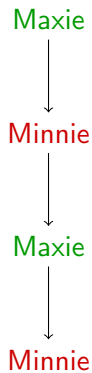
Maxie



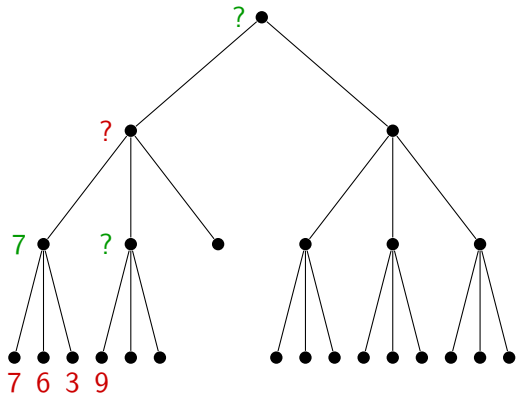
Minnie



Minimax Search



Minimax Search



Minimax Search

Maxie



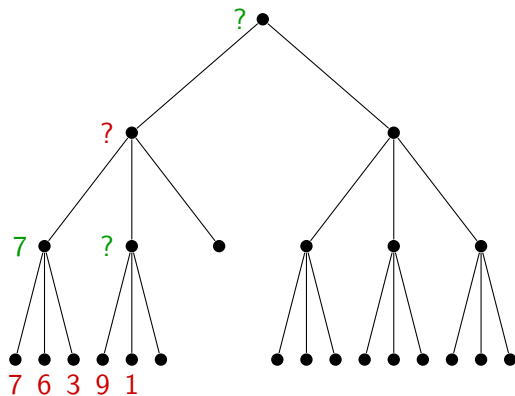
Minnie



Maxie



Minnie



Minimax Search

Maxie



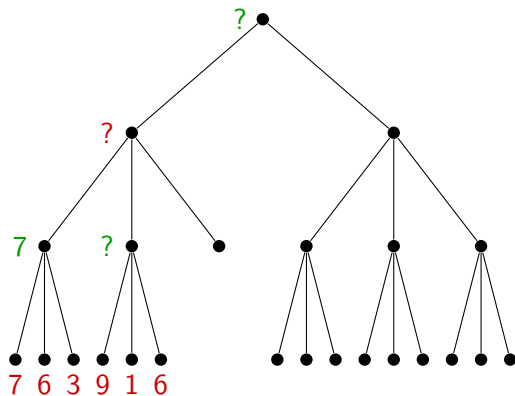
Minnie



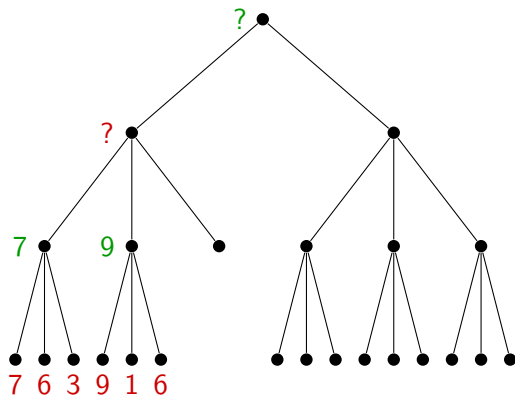
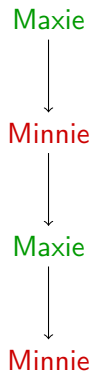
Maxie



Minnie



Minimax Search



Minimax Search

Maxie



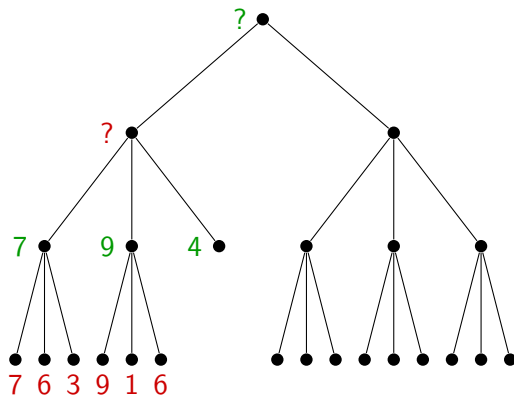
Minnie



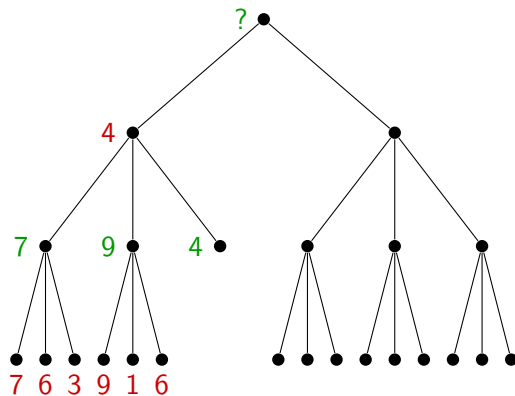
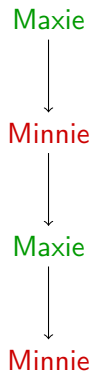
Maxie



Minnie



Minimax Search



Minimax Search

Maxie



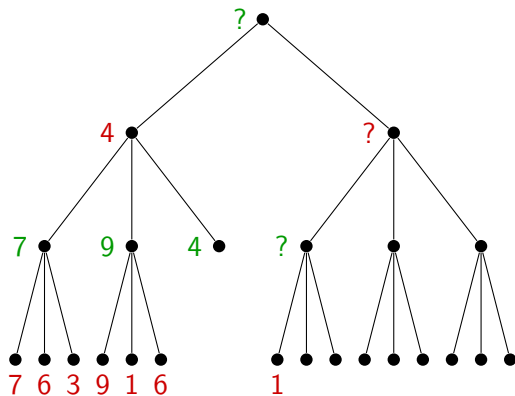
Minnie



Maxie



Minnie



Minimax Search

Maxie



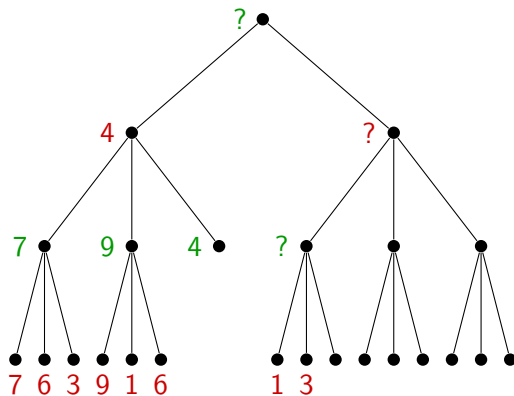
Minnie



Maxie



Minnie



Minimax Search

Maxie



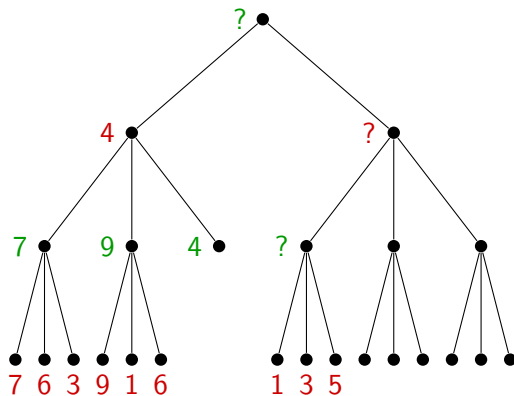
Minnie



Maxie



Minnie



Minimax Search

Maxie



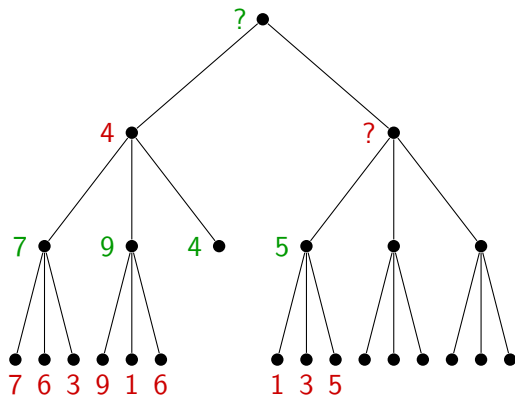
Minnie



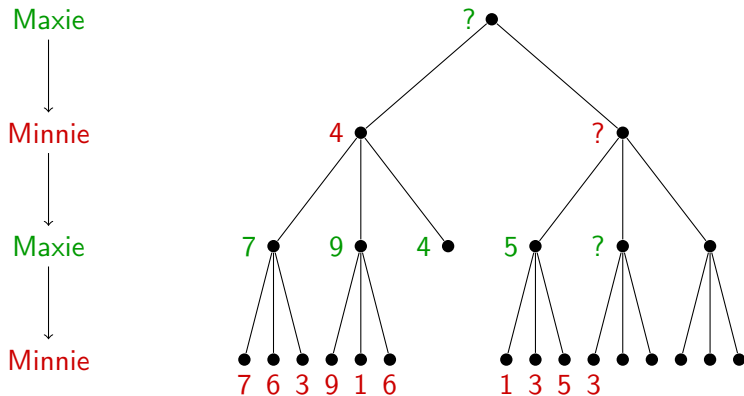
Maxie



Minnie



Minimax Search



Minimax Search

Maxie



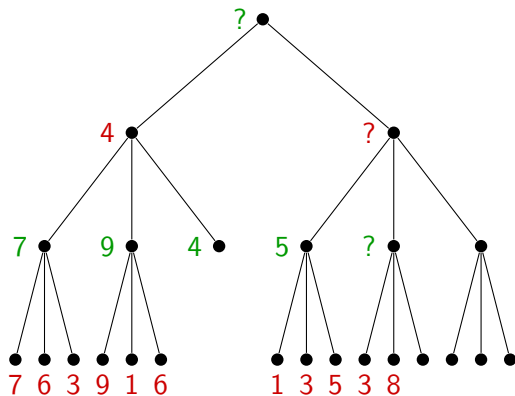
Minnie



Maxie



Minnie



Minimax Search

Maxie



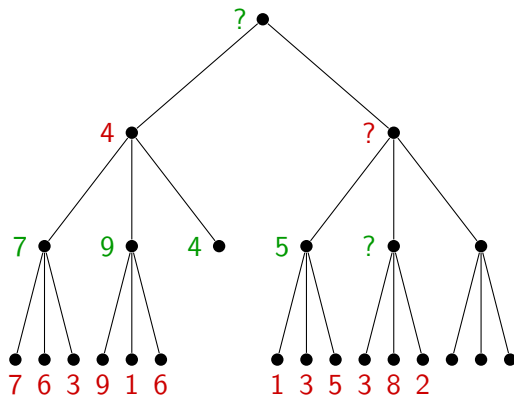
Minnie



Maxie



Minnie



Minimax Search

Maxie



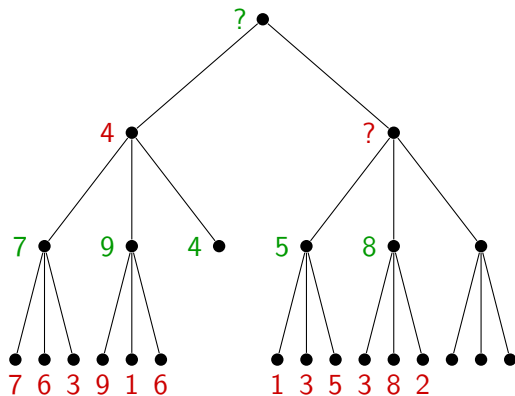
Minnie



Maxie



Minnie



Minimax Search

Maxie



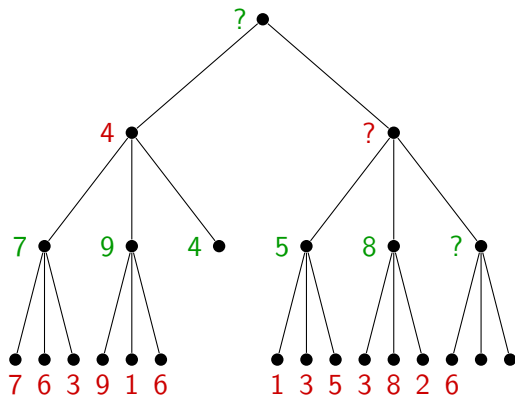
Minnie



Maxie



Minnie



Minimax Search

Maxie



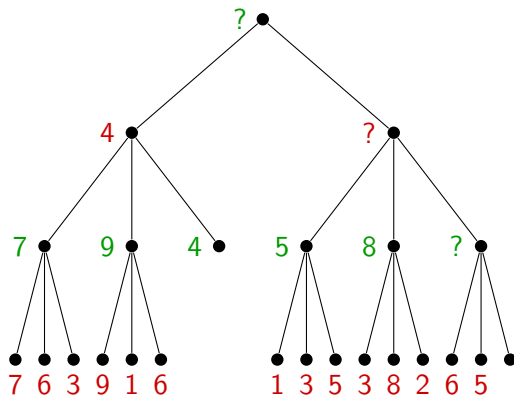
Minnie



Maxie



Minnie



Minimax Search

Maxie



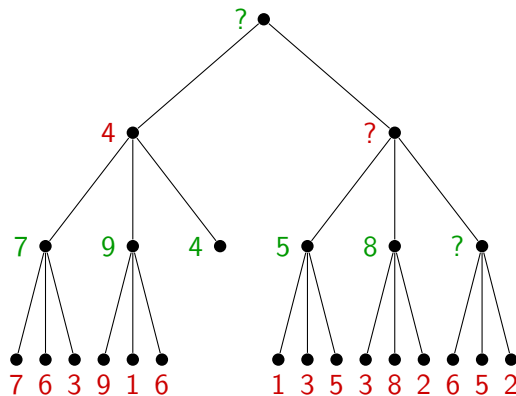
Minnie



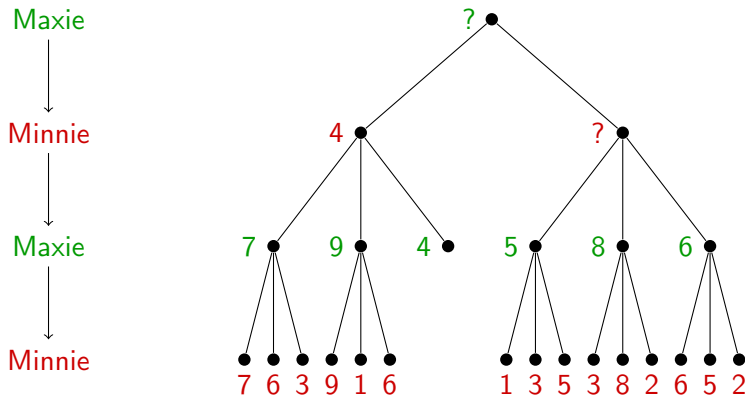
Maxie



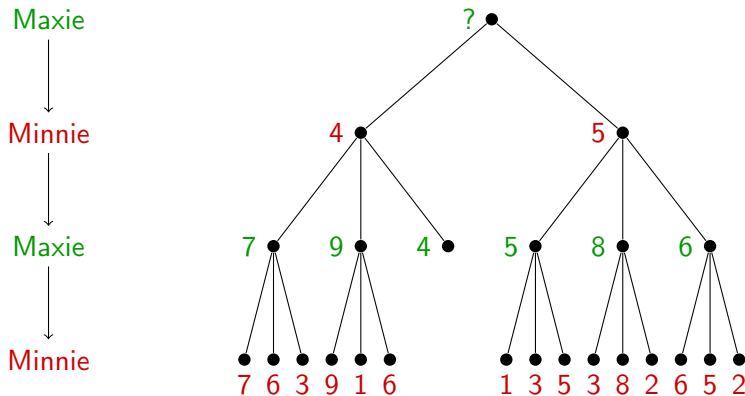
Minnie



Minimax Search



Minimax Search



Minimax Search

Maxie



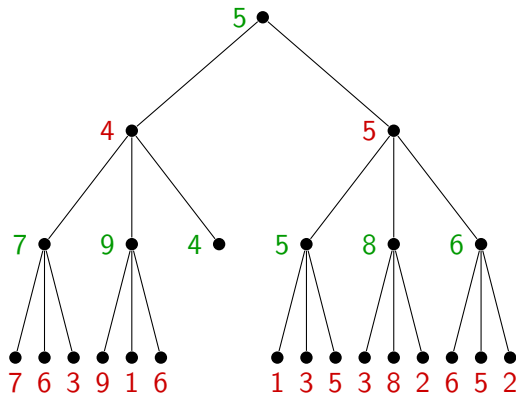
Minnie



Maxie



Minnie



Alpha/Beta Search

Maxie



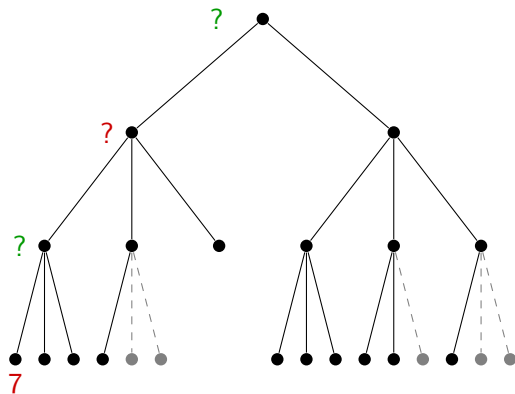
Minnie



Maxie



Minnie



Alpha/Beta Search

Maxie



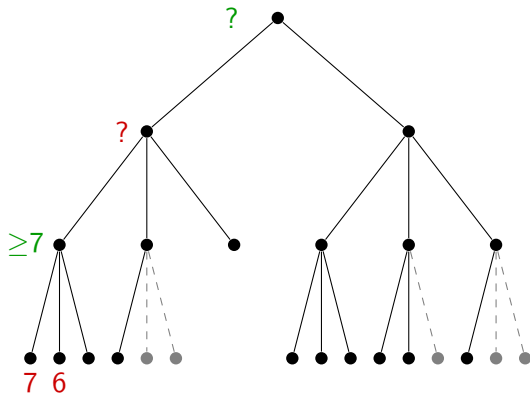
Minnie



Maxie



Minnie



Alpha/Beta Search

Maxie



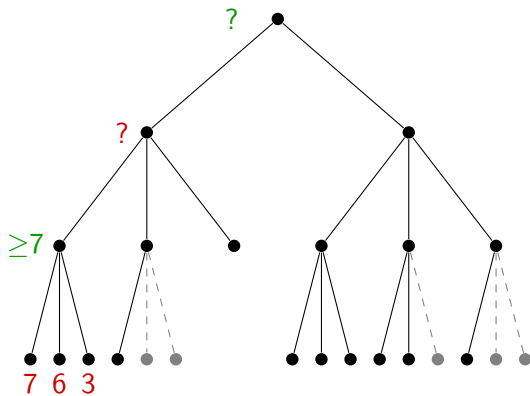
Minnie



Maxie



Minnie



Alpha/Beta Search

Maxie



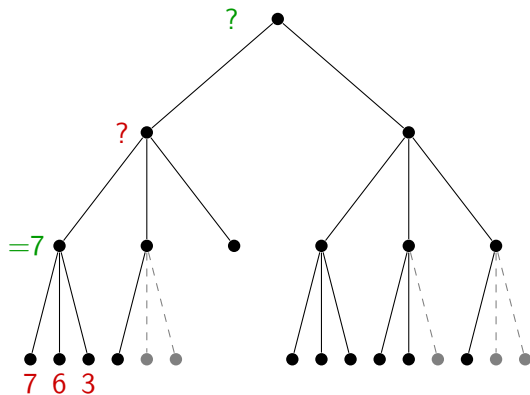
Minnie



Maxie



Minnie



Alpha/Beta Search

Maxie



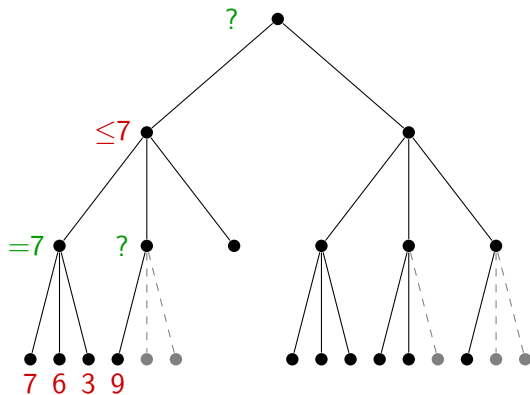
Minnie



Maxie



Minnie



Alpha/Beta Search

Maxie



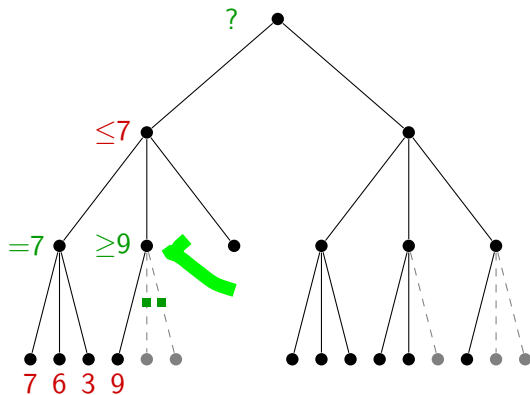
Minnie



Maxie



Minnie



Alpha/Beta Search

Maxie



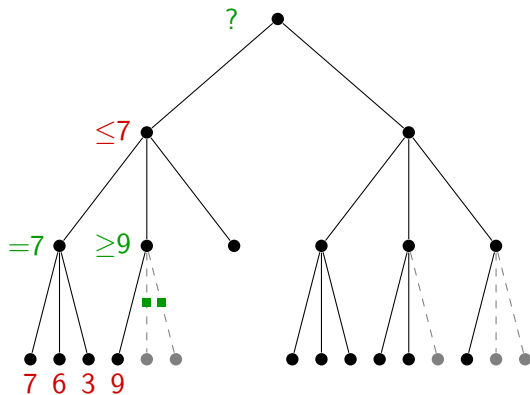
Minnie



Maxie



Minnie



Alpha/Beta Search

Maxie



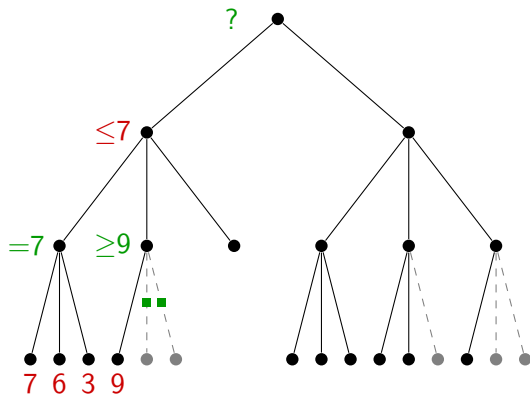
Minnie



Maxie



Minnie



Alpha/Beta Search

Maxie



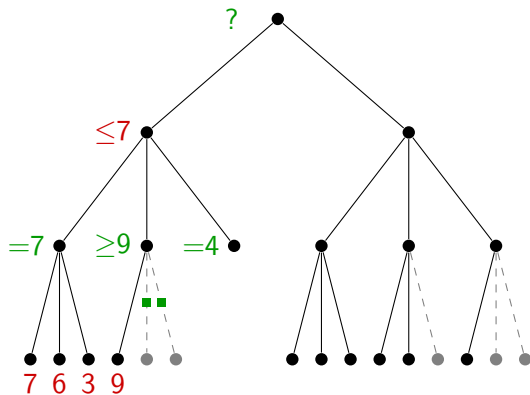
Minnie



Maxie



Minnie



Alpha/Beta Search

Maxie



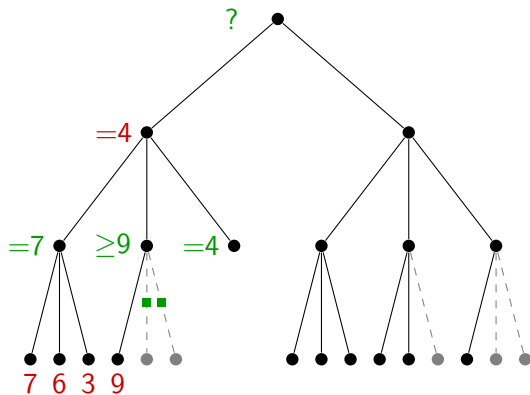
Minnie



Maxie



Minnie



Alpha/Beta Search

Maxie



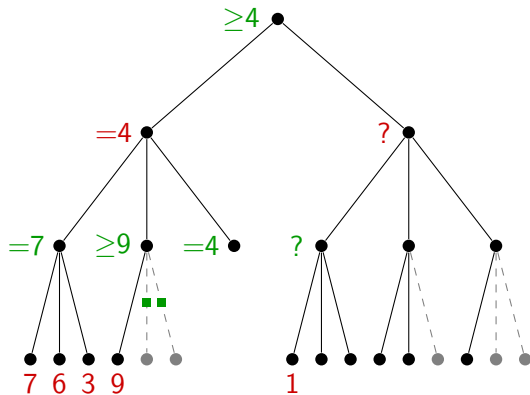
Minnie



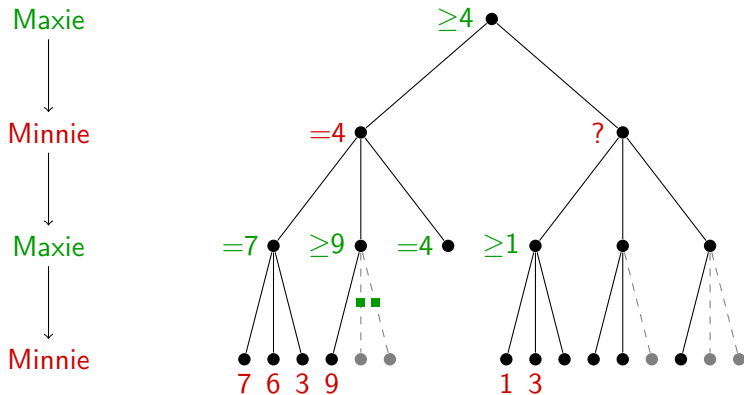
Maxie



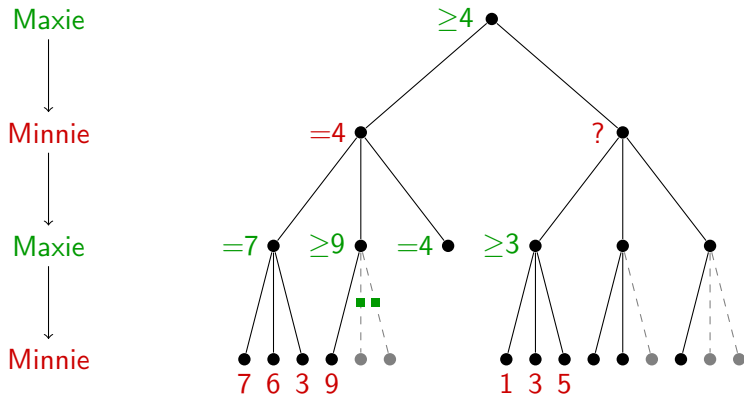
Minnie



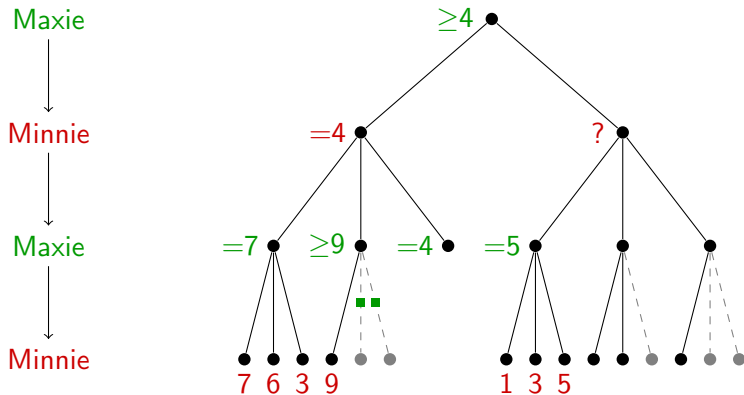
Alpha/Beta Search



Alpha/Beta Search



Alpha/Beta Search



Alpha/Beta Search

Maxie



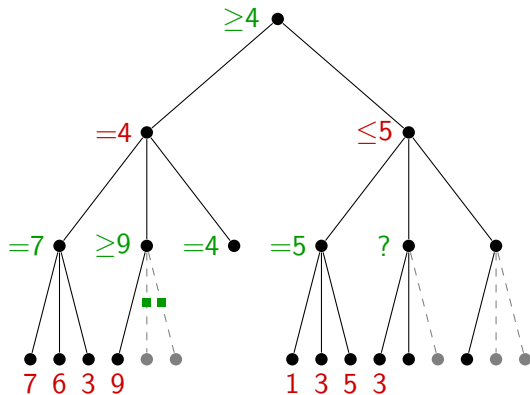
Minnie



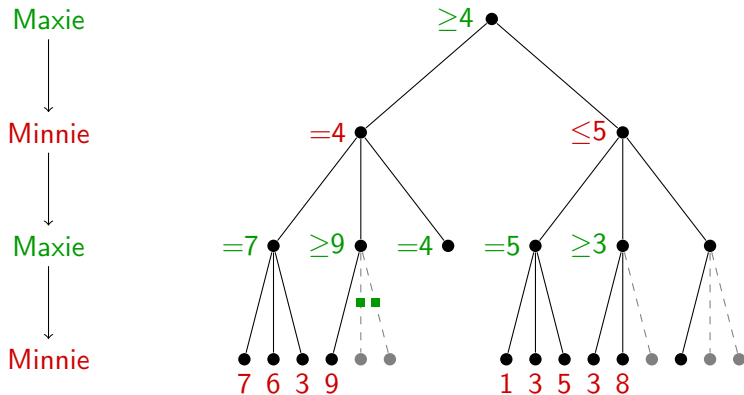
Maxie



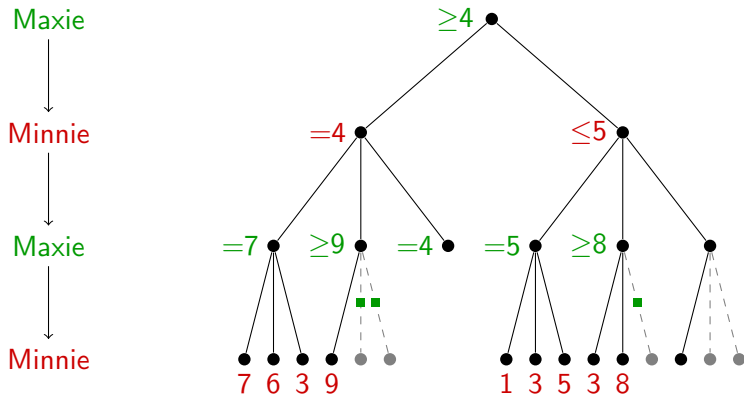
Minnie



Alpha/Beta Search

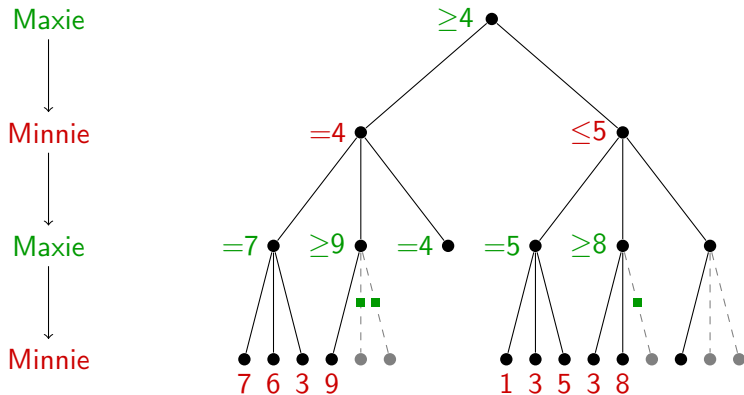


Alpha/Beta Search



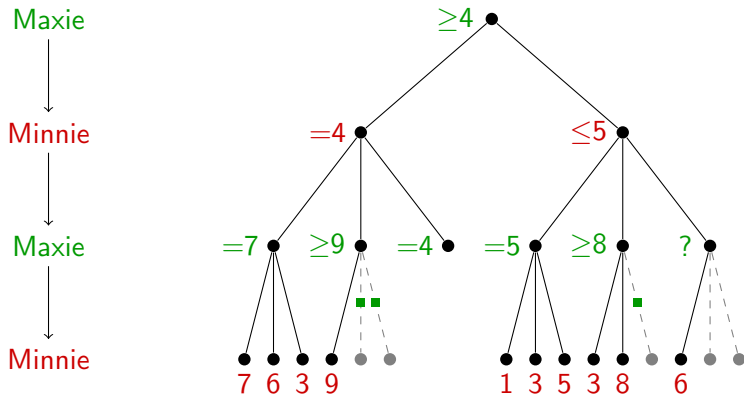
- We need to pass information *down* the tree during search
 - **Minnie** sets an *upper bound* (e.g., ≤ 5)
 - **Maxie** sets a *lower bound* (e.g., ≥ 4)

Alpha/Beta Search



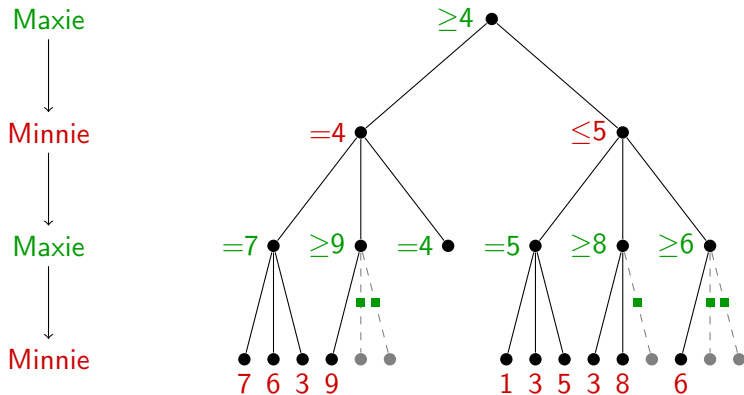
- We need to pass information *down* the tree during search
 - **Minnie** sets an *upper bound* (e.g., ≤ 5)
 - **Maxie** sets a *lower bound* (e.g., ≥ 4)

Alpha/Beta Search



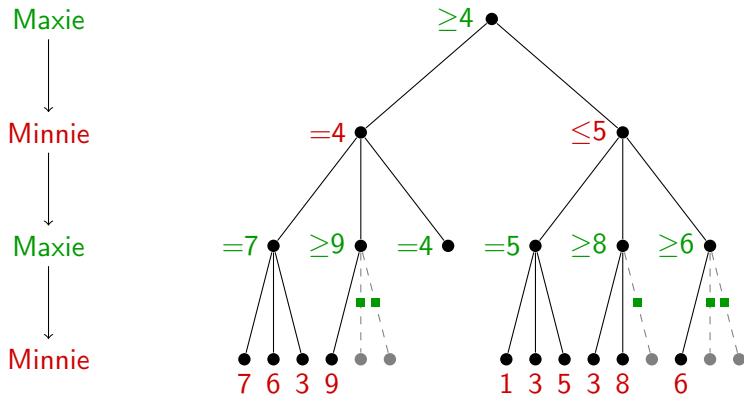
- We need to pass information *down* the tree during search
 - Minnie sets an *upper bound* (e.g., ≤ 5)
 - Maxie sets a *lower bound* (e.g., ≥ 4)

Alpha/Beta Search



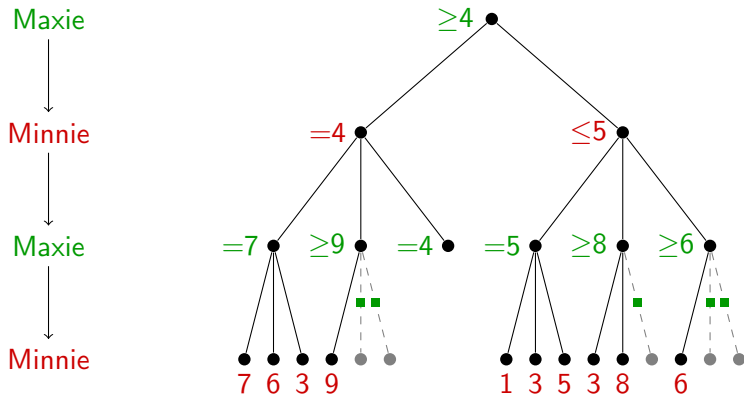
- We need to pass information *down* the tree during search
 - **Minnie** sets an *upper bound* (e.g., ≤ 5)
 - **Maxie** sets a *lower bound* (e.g., ≥ 4)

Alpha/Beta Search



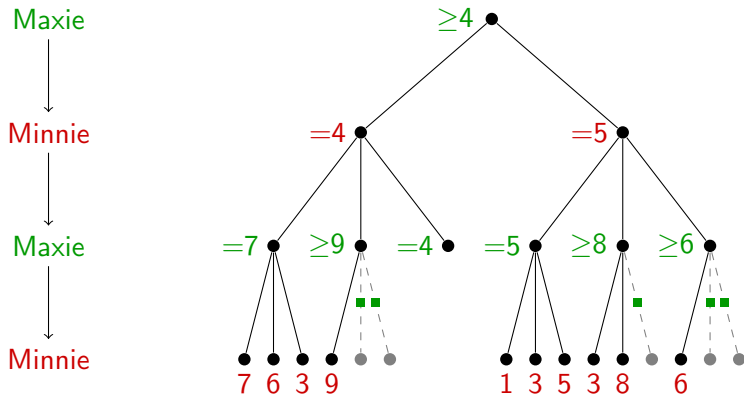
- We need to pass information *down* the tree during search
 - **Minnie** sets an *upper bound* (e.g., ≤ 5)
 - **Maxie** sets a *lower bound* (e.g., ≥ 4)

Alpha/Beta Search



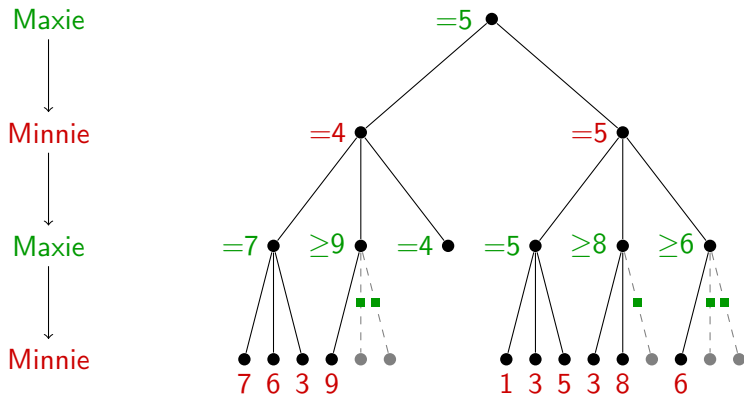
- We need to pass information *down* the tree during search
 - Minnie sets an *upper bound* (e.g., ≤ 5)
 - Maxie sets a *lower bound* (e.g., ≥ 4)
- Pass interval (α, β) !
 - Maxie cuts off search if value is greater than β
 - Minnie cuts off search if value is less than α

Alpha/Beta Search



- We need to pass information *down* the tree during search
 - Minnie sets an *upper bound* (e.g., ≤ 5)
 - Maxie sets a *lower bound* (e.g., ≥ 4)
- Pass interval (α, β) !
 - Maxie cuts off search if value is greater than β
 - Minnie cuts off search if value is less than α

Alpha/Beta Search

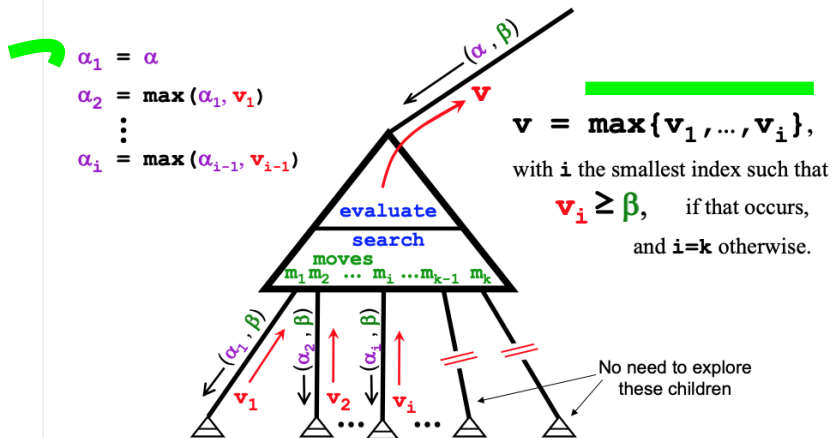


- We need to pass information *down* the tree during search
 - Minnie sets an *upper bound* (e.g., ≤ 5)
 - Maxie sets a *lower bound* (e.g., ≥ 4)
- Pass interval (α, β) !
 - Maxie cuts off search if value is greater than β
 - Minnie cuts off search if value is less than α

Alpha/Beta Pruning

- We pass interval (α, β) down during search (with $\alpha < \beta$)
- α is the best (largest) Maxie can achieve (so far)
- β is the best (smallest) Minnie can achieve (so far)
 - If Maxie sees a move to a node with value $v \geq \beta$ stop searching from current node
 - Minnie would never choose the current node, because it can already do better
- Conversely:
 - If Minnie sees a move to a node with value $v \leq \alpha$ stop search from the current node
 - Maxie would never choose the current node, because it can already do better

Alpha-Beta at a **Maxie** Node



- Sometimes, optimization **enhance parallelism**

Parallelism Revisited

- Sometimes, optimization **enhance parallelism**
 - From insertion sort to merge sort

Parallelism Revisited

- Sometimes, optimization **enhance parallelism**
 - From insertion sort to merge sort
- Sometimes, optimization **reduce parallelism**

Parallelism Revisited

- Sometimes, optimization **enhance parallelism**
 - From insertion sort to merge sort
- Sometimes, optimization **reduce parallelism**
 - From minimax to alpha/beta game tree search

Summary

- Bounding search with estimators
- Review minimax search
- Alpha/beta pruning