CSL020U4E: Artificial Intelligence
Lecture-07 (Board Game Algorithms)

Sumit Kumar Pandey

November, 2025

GamePlay *Algorithm*

GamePlay(MAX)

while game not over

call *k*-ply **search**

make move

get MIN's **move**

Minimax *Algorithm*

The algorithm Minimax does a depth first search of the game tree. This version recursively calls itself to compute the value of each child. A node is a terminal node if it is on the horizon.

Minimax(N)
  if N **is a terminal node**
    $value \leftarrow$ Evaluate(N)
  else if N **is a** MAX **node**
    value $\leftarrow -LARGE$
    for each child C **of** N
      $value \leftarrow$ **max**($value$, Minimax(C))
  else $value \leftarrow +LARGE$
    for each child C **of** N
      $value \leftarrow$ **min**($value$, Minimax(C))
  return $value$

AlphaBeta *Pruning*

AlphaBeta(N, α, β)
  if N **is a terminal node**
    return Evaluate(N)
  if N **is a** MAX **node**
    for each child C **of** N
      $\alpha \leftarrow$ **max**$(\alpha,$ AlphaBeta$(C, \alpha, \beta))$
      if $\alpha \geq \beta$ **then return** β
    return α
  else $\triangleright N$ **is a** MIN **node**
    for each child C **of** N
      $\beta \leftarrow$ **min**$(\beta,$ AlphaBeta$(C, \alpha, \beta))$
      if $\alpha \geq \beta$ **then return** α
    return β

Thank You