Minimax Pseudocode

MiniMax(Board)
  best.mv = [not yet defined]
  best.score = -9999
  For each legal move m
  {  
    make move m.mv on Board
    m.score = MIN
    if (m.score > best.score) then best = m
    retract move m.mv on Board
  }
  Make move best.mv

MINIMAX

MAX
  if (game over) return EVAL-ENDING
  else if (max depth) return EVAL
  else
    best.score = -9999
    for each computer legal move m
    {  
      make move m.mv on Board
      m.score = MIN
      if (m.score > best.score) then best = m
      retract move m.mv on Board
    }
  return best.score

MIN
  if (game over) return EVAL-ENDING
  else if (max depth) return EVAL
  else
    best.score = 9999
    for each human legal move m.mv
    {  
      make move m.mv on Board
      m.score = MAX
      if (m.score < best.score) then best = m
      retract move m.mv on Board
    }
  return best.score

Assumptions:
- EVAL applies heuristics to evaluate position B, from computer’s perspective (more+ = computer winning, more- = human winning).
- EVAL-ENDING = +999 (computer won), -999 (human won), 0 if drawn.
- Board (game state) is global. All other variables are local.
- Variables “best” and “m” represent legal moves, where:
  - “.mv” represents the move itself
  - “.score” represents the evaluation for the move

Useful websites:
  http://ai-depot.com/articles/minimax-explained/
  http://en.wikipedia.org/wiki/Claude_Elwood_Shannon