LAB-10

The Statement.

As per the law, it is a crime for an American to sell weapons to hostile nations, Country A, an enemy of America, has some missiles were sold to it by Robert, who is an American citizen.

To Prove

Robert is a criminal

Forward Chaining Tree

$\exists x\, Owns(A,x) \land Missile(x)$

$Owns(A,T_1)$

$Missile(T_1)$

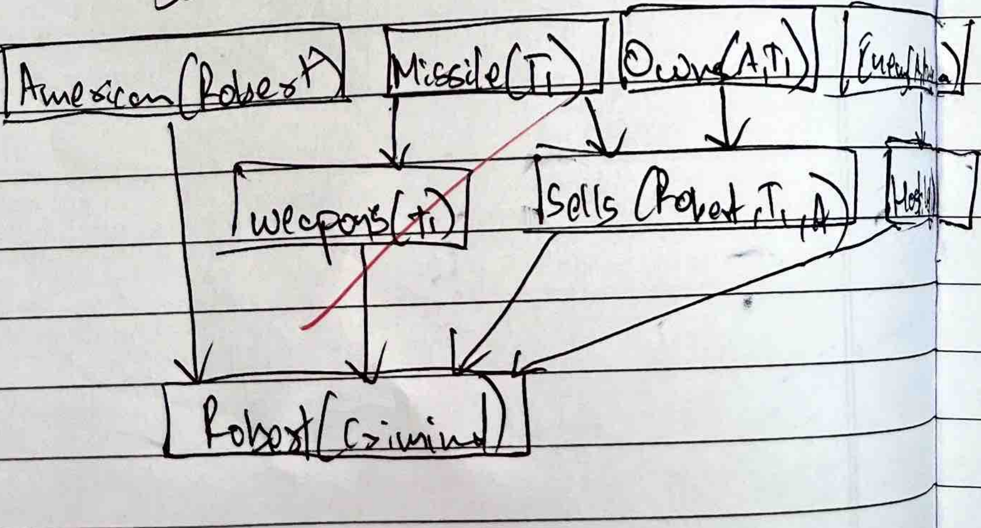
$\forall x\, Missile(x) \land Owns(A,x) \Rightarrow Sells(Robert,x,A)$

$American(Robert)$

$Missile(x) \Rightarrow Weapon(x)$

$Enemy(A, America)$

$\forall x\, Enemy(x, America) \Rightarrow Hostile(x)$



Min-Max Algorithm for Tix-Tac-Toe

```
function ALPHA-BETA-SEARCH(state) returns an action
 $v \leftarrow$ MAX-VALUE (state, $-\infty$, $+\infty$)
 return the action in ACTIONS (state) with value $v$

function MAX-VALUE(state, $\alpha$, $\beta$) returns a utility value
  if TERMINAL-TEST (state) then return UTILITY(state
   $v \leftarrow -\infty$
   for each a in ACTIONS (state) do
      $v \leftarrow$ MAX ($v$, MIN-VALUE(RESULT(s,a), $\alpha$, $\beta$
      if $v \geq \beta$ then return $v$
      $\alpha \leftarrow$ MAX($\alpha$, $v$)
   return $v$

function MIN-VALUE (state, $\alpha$, $\beta$) returns a utility value
   if TERMINAL-TEST(state) then return UTILITY(state
   $v \leftarrow +\infty$
   for each a in ACTIONS (state) do
      $v \leftarrow$ MIN ($v$, MAX-VALUE (RESULT(s,a), $\alpha$, $\beta$))
      if $v \leq \alpha$ then return $v$
      $\beta \leftarrow$ Min($\beta$, $v$)
   return $v$.
```

Output

```
 O |   |        Enter row and column: 0 0
---+---+---     AI is making a move...
   |   |
---+---+---
   |   |

 O | X |        Enter row and column: 1 2
---+---+---     AI is making a move...
   |   |

 O | X | X      Enter row and column: 1 0
---+---+---     AI is making a move...
   |   | O
---+---+---
   |   |

 O | X | X      Enter row and column: 1 1
---+---+---
 O |   | O      Human Wins!
---+---+---
 X |   |

 O | X | X
---+---+---
 O | O | O
---+---+---
 X |   |
```

Alpha-Beta Pruning for 8-Queens.

```
function ~~8Q~~ queens (board, row, alpha, max_player)
    IF row == N THEN
        RETURN "SUCCESS"
    FOR col in 0 To N-1 DO
        IF is_safe (board, row, col) THEN
            Place queen at (row, col)
            result ← queens(board, row+1, alpha, beta,
                            NOT Max_player)

            IF result == "SUCCESS" THEN
                RETURN "SUCCESS"

            Remove queen from (row, col)
        IF max_player THEN
            alpha ← MAX(alpha, current_score)
        ELSE
            beta ← MIN(beta, current_score)
        IF alpha >= beta THEN
            BREAK
    RETURN "FAILURE"
```

Output

Solution

```
Q . . . . . . .
. . . . Q . . .
. . . . . . . Q
. . . . . Q . .
. . Q . . . . .
. . . . . . Q .
. Q . . . . . .
. . . Q . . . .
```

SnehalB
3/12/24