Mandatory Exercise - Simple Tic-Tac-Toe

Anders Wind - Awis@itu.dk

Task 1

The miniMax-Descision function should be called when Max has a possibility to choose an action.
Therefore the input state must be a state where the coin has just been flipped.

```
// this function is supposed to be the same as in the book.
function MiniMax-Descision(state) return action
    return arg max(a belongs to Actions(s)) MinValue(Result(state,a))

// the rest of these functions are modified.
function Max-Value(state) return a utility value
    if Terminal-Test(state) then return Utility(state)

    v <- inf*-1
    foreach a in Action(state) do
        v <- Max(v,Chance-Min(Result(state,a))
    return v

function Min-Value(state) return a utility value
    if Terminal-Test(state) then return Utility(state)
    v <- inf
    foreach a in Action(state) do
        v <- Min(v,Chance-Max(Result(state,a))
    return v

function Chance-Min(state)
    return (Min-Value(Result(state, head)), Min-Value(Result(state, tails)))*0.5

function Chance-Max(state)
    return (Max-Value(Result(state, head)), Max-Value(Result(state, tails)))*0.5
```
Note that this solution only works for the game given in the assignment. We start with a throw

Task 2

a) See the attached image "Simple-TTT-GameTree.png" to see the GameTree. For completeness the
subtree where tails was the first value of the flipped coin, is included but should be disregarded for this task.

b) On the game tree "Simple-TTT-GameTree.png" the evaluation of a ExpectMiniMax is run and the highest
utility, 0.5, is achieved by placing the x in the first column, second row. So that is the decision for MAX's
first move if the coin flipped to HEAD.