Monte Carlo Tree Search

Algorithm 1 Monte Carlo Tree Search

```
function MCTS($s_{root}$)
    while within computational budget do
        $s \leftarrow$TreePolicy($s_{root}$)
        winner$\leftarrow$DefaultPolicy($s$)
        Backup(s, winner)
    end while
    return Action($\text{argmax}_{s' \in \text{children}(s_{root})} \frac{Q(s')}{N(s')}$)
end function

function TreePolicy($s$)
    while $s$ is not terminal do
        if $s$ is not fully expanded then
            return Expand(s)
        else
            $s \leftarrow$ BestChild(s)
        end if
    end while
    return $s$
end function

function Expand($s$)
    child $\leftarrow$ previously unexpanded child of $s$
    Update the tree with $(s, child)$
    return child
end function

function BestChild($s$)
    return $\text{argmax}_{s' \in \text{children}(s)} \left( \frac{Q(s')}{N(s')} + c\sqrt{\frac{\ln N(s)}{N(s')}} \right)$
end function

function DefaultPolicy($s$)
    while $s$ is not terminal do
        $s \leftarrow$ random child of $s$
    end while
    return winner
end function

function Backup(s, winner)
    while $s$ is not Null do                              ▷ Parent of root is Null
        $N(s) \leftarrow N(s) + 1$
        $Q(s) \leftarrow Q(s) + \Delta(s, winner)$        ▷ Update winning count based on who's in control of $s$
        $s \leftarrow$parent($s$)
    end while
end function
```

In the simplest form, $\Delta(s, winner)$ is 1 if the player **in control of** s (i.e., s is the outcome of that player's action) has won in the rollout, and 0 otherwise.