MiniMax

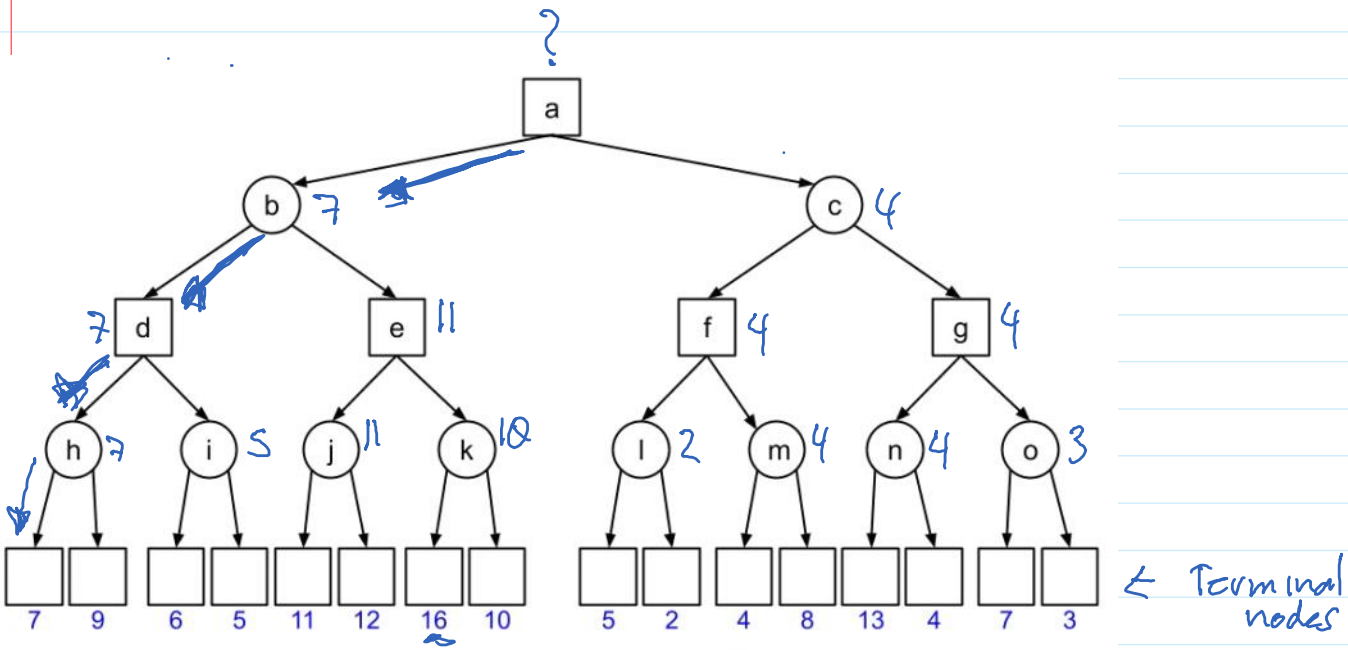
Friday, April 3, 2020 2:29 AM

2-agents

Perfect information game { fully-observable, deterministic, turn-based, zero-sum

EXAMPLE:

MAX □ MIN ○ minimax(s)



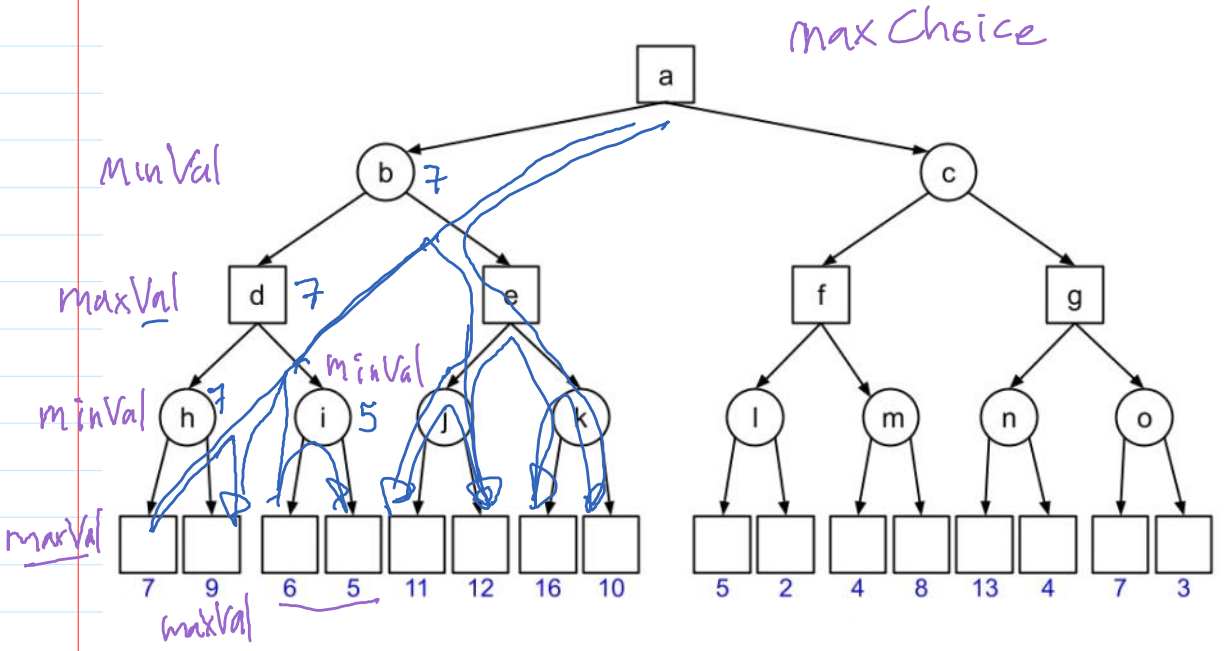
- Algorithm:

```
actions( s : state )          :- returns all actions possible in state s
term( s : state )             :- true: s is terminal
utility( s : state )          :- utility of terminal states
result( s : state , a : action ) :- returns resulting state of
                                    applying action a on state s.
```

maxValue :- } value of nodes owned by max/min.
minValue :- }

```
PROCEDURE maxChoice( s0 : state )
  FOREVERY a in actions(s0) :
    besta := a that maximizes minValue( result( s0, a ) )
  RETURN besta;

PROCEDURE minValue( s : state )
  IF term( s ) THEN RETURN utility( s )
  v := ∞
  FOREACH a in actions( s ) DO
    v := MIN( v, maxValue( result( s, a) ) )
  RETURN v
      .
PROCEDURE maxValue( s : state )
  IF term( s ) THEN RETURN utility( s )
  v := -∞
  FOREACH a in actions( s ) DO
    v := MAX( v, minValue( result( s, a) ) )
  RETURN v
```



Recursive DFS