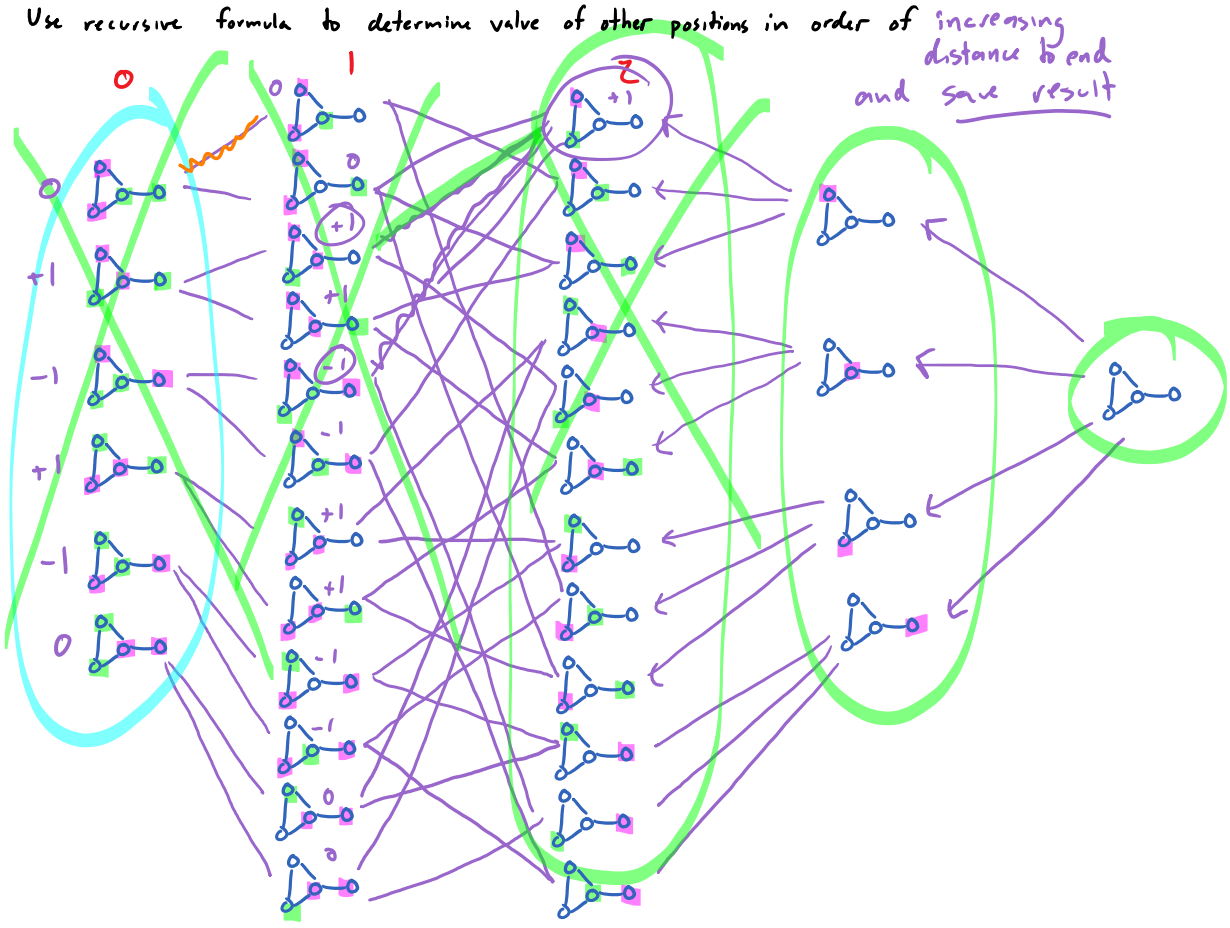
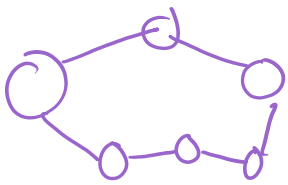
Dynamic Programming

Order positions by maximum distance to end.

Determine winner of distance 0 positions (terminal) by referring to rules

Use recursive formula to determine value of other positions in order of increasing distance to end and save result



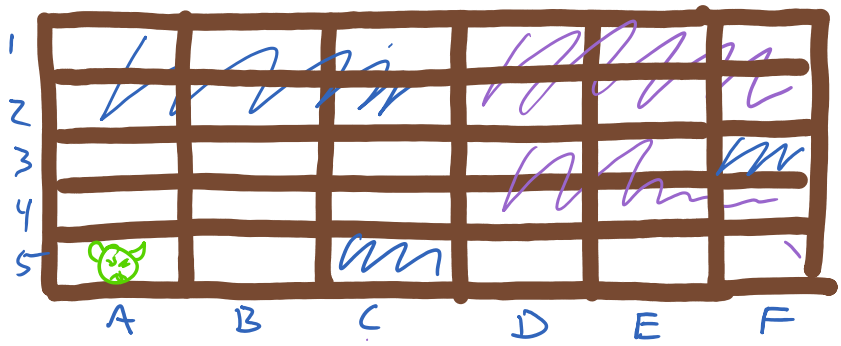


Play on m×n grid. Take turns selecting remaing cell, remove all above and to right

Last move loses.

misere

impartial

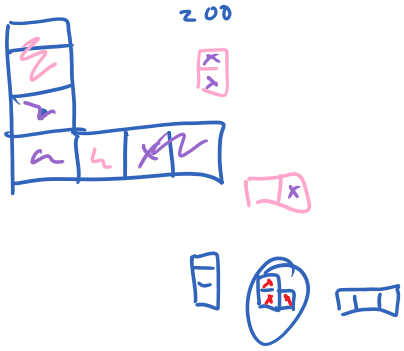


N: next player has winning move

P: prev. player has winning strategy

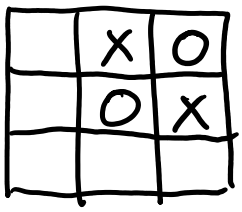
```
outcome_class(p)
if p is end of game
    return value according to rules
else
    S ← positions reachable in 1 move from p
    if S contains a P position
        return N
    else
        return P
```

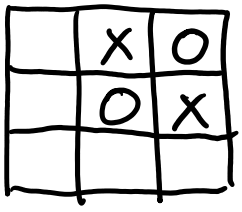
000
100
110
200
111
210
211
220
221
222

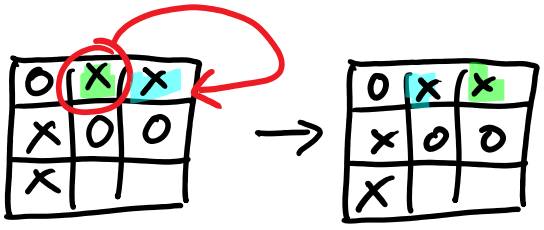


Quixo

Quixo | Board Game | BoardGameGeek

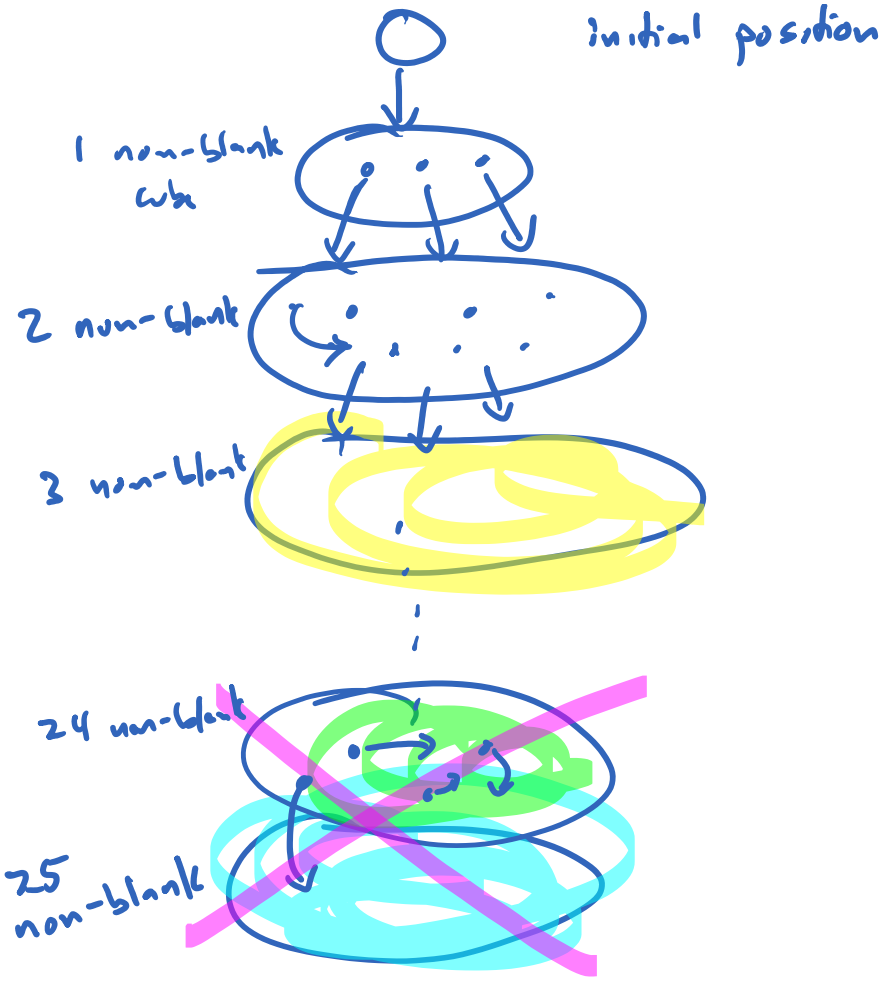






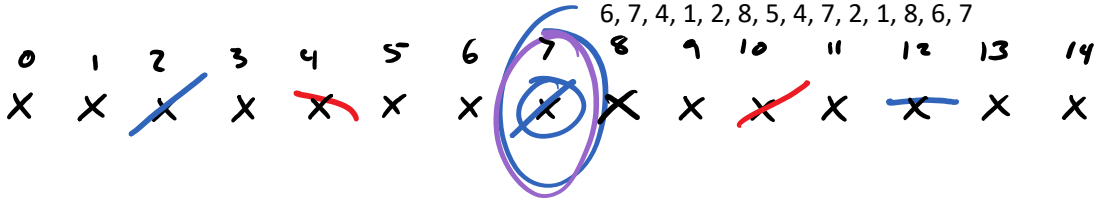
Value Iteration

mark all positions as draws
mark all terminal positions according to rules
repeat
  for all positions p marked as draws
    if some successor is win for current player @ p
      mark p as win for current player @ p
    else if all successors marked as losses
      mark p as win for other player
until no new positions marked


initial position
1 non-blank cube
2 non-blank
3 non-blank
24 non-blank
25 non-blank

Kayles

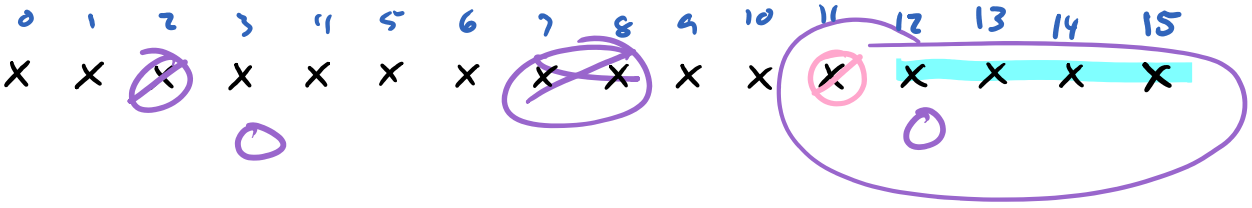
0, 1, 2, 3, 1, 4, 3, 2, 1, 4, 2, 6, 4, 1, 2, 7, 1, 4, 3, 2, 1, 4, 6, 7, 4, 1, 2, 8, 5, 4, 7, 2, 1, 8, 6, 7



Start with row of n pins

On each turn, take 1 or 2 adjacent pins

If no possible moves, you lose



Game Positions

Game position = set of positions one can move to

In traditional 1-row Nim

0 stones $\sqcup = \{\} = \emptyset$

$0 = \{\sqcup\}$

$00 = \{\sqcup, 0\}$

$000 = \{\sqcup, 0, 00\}$

$0000 =$

$\vdots$