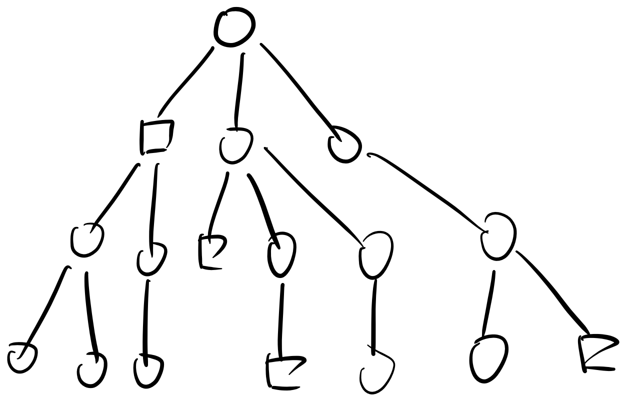


O: good

□: bad

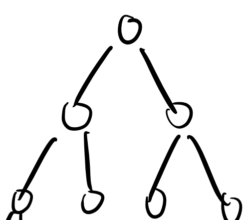
find a good path from the root to a leaf.

all nodes are goods

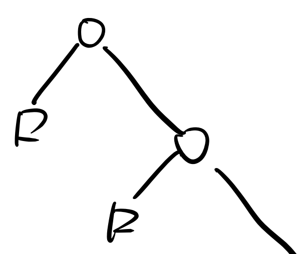
Dfs(u)

1. if u is bad,
   return None
2. else if u has no children. //u is a good leaf.
   return u
3. else //u is a good internal node
   for each good child v of u,
      path = Dfs(v)
      if path ≠ None:
         return u ⟶ path

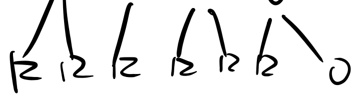
backtracking = dfs + pruning

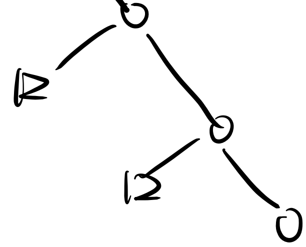


O(n)



O(h)





n queens problem

Given a $n \times n$ chessboard,

find a feasible placement of n queens

↓ feasible: no two of them can attack each other

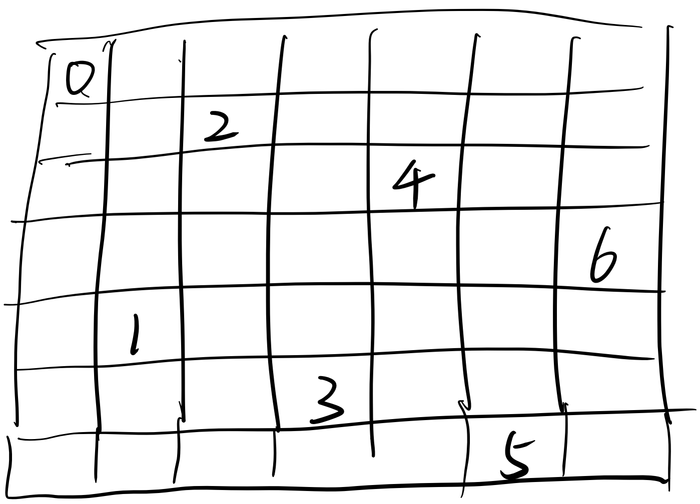
↓ attack: same row, column, or diagonal.

Facts:

1. for $n > 3$, a feasible placement ~~must~~ always exists.
2. for some special n (prime, $6k+1$, $6k+5$ ...)

   a feasible placement can be found efficiently

$n=7$

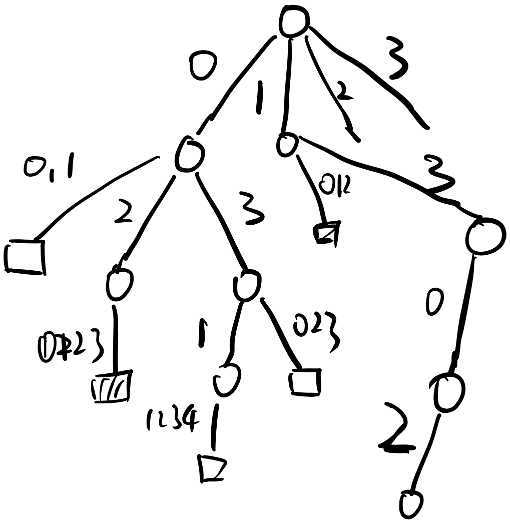


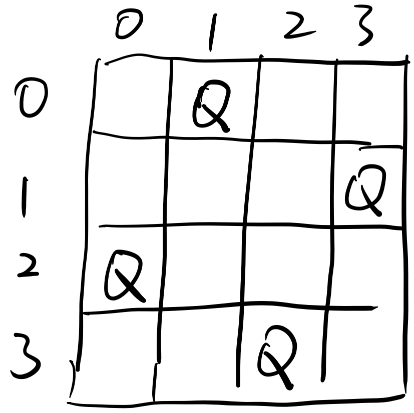
3. for general n,

   it is NP-hard to find a feasible placement.

bruteforce. $O(n! \cdot n^2)$

backtracking $O(n! \cdot n)$





$P[i]$ = the position at ith row for $i=0, \ldots, n-1$

```
NQ(n):
  P[i] = -1 for i=0 to n-1
  i=0
  while i>=0 & i<n
      next_choice = -1
      for j = P[i]+1 to n-1
          if j cannot attack P[0], ..., P[i-1]
              next_choice = j
              break
      if next_choice == -1
          P[i] = -1
          i = i-1
      else
         P[i] = next_choice
```

$i = i + 1$

if $i == -1$:

  no feasible placement

else // $i == n$

  P is a feasible placement

Turnpike problem

$A = \{0, 1, 2\}$

$D = \{1, 1, 2\}$

$A = \{0, 1, 4\}$

$D(A) = \{1, 3, 4\}$

$|D(A)| = \binom{|A|}{2} = \frac{|A|(|A|-1)}{2}$

multiset

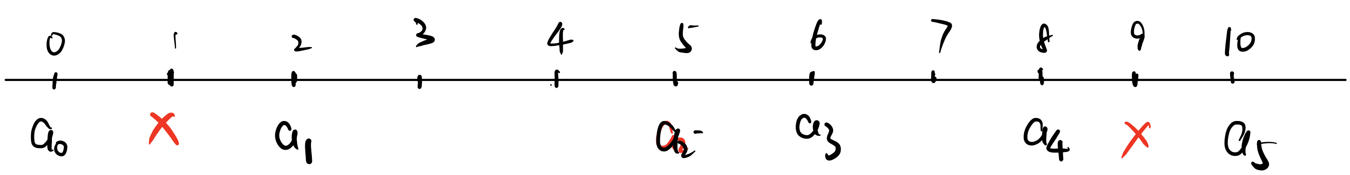
Given D, find a $A = \{a_0 \le a_1 \le \cdots \le a_{n-1}\}$ (assume $a_0 = 0$)

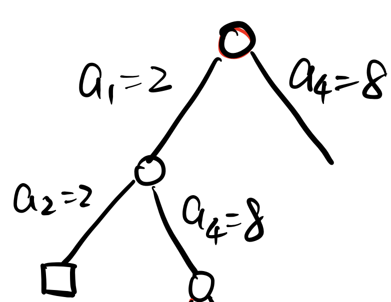
s.t. $D(A) = D$

$A =$

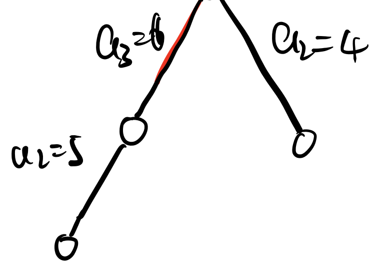
$|D| = 15$

$D = \{\cancel{1}, \cancel{2}, \cancel{2}, \cancel{2}, \cancel{3}, \cancel{3}, \cancel{4}, \cancel{4}, \cancel{5}, \cancel{5}, \cancel{6}, \cancel{6}, \cancel{8}, \cancel{8}, \cancel{10}\} \Rightarrow |A| = 6$





$A = \{0, 2, 5, 6, 8, 10\}$



maximum distance remaining in D must result from either $a_{n-1} - a_i$

or $a_i - a_0$

$D = \{$ distance involving unknow points $\}$

Input: a multiset D

Output: a multiset A.

$A = \{0, \max(D)\}$

remove $\max(D)$ from D

TP(D, A)

TP(D, A):

  if $D == \phi$: $O(1)$

    return true

  $a = \max(D)$ $O(h) = O(\lg n^2) = O(\lg n)$

  $a' = \max(A) - a$ $O(n)$

  for $x = a$ or a' $\rightarrow O(n)$

$|D| \le n^2/2$

$|A| \le n$

$|\Delta| \le n$

$O(n)$

$\Delta = dist(x, A)$ // compute distance between x and every points in A

if $\Delta \subseteq D$: $O(n \cdot \lg n)$

remove Δ from D $O(n \cdot \lg n)$

add x to A $O(1)$

If TP(D, A):

return True

else

add Δ to D $O(n \cdot \lg n)$

remove x from A $O(n)$

return false

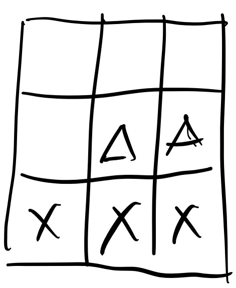
Balanced BST

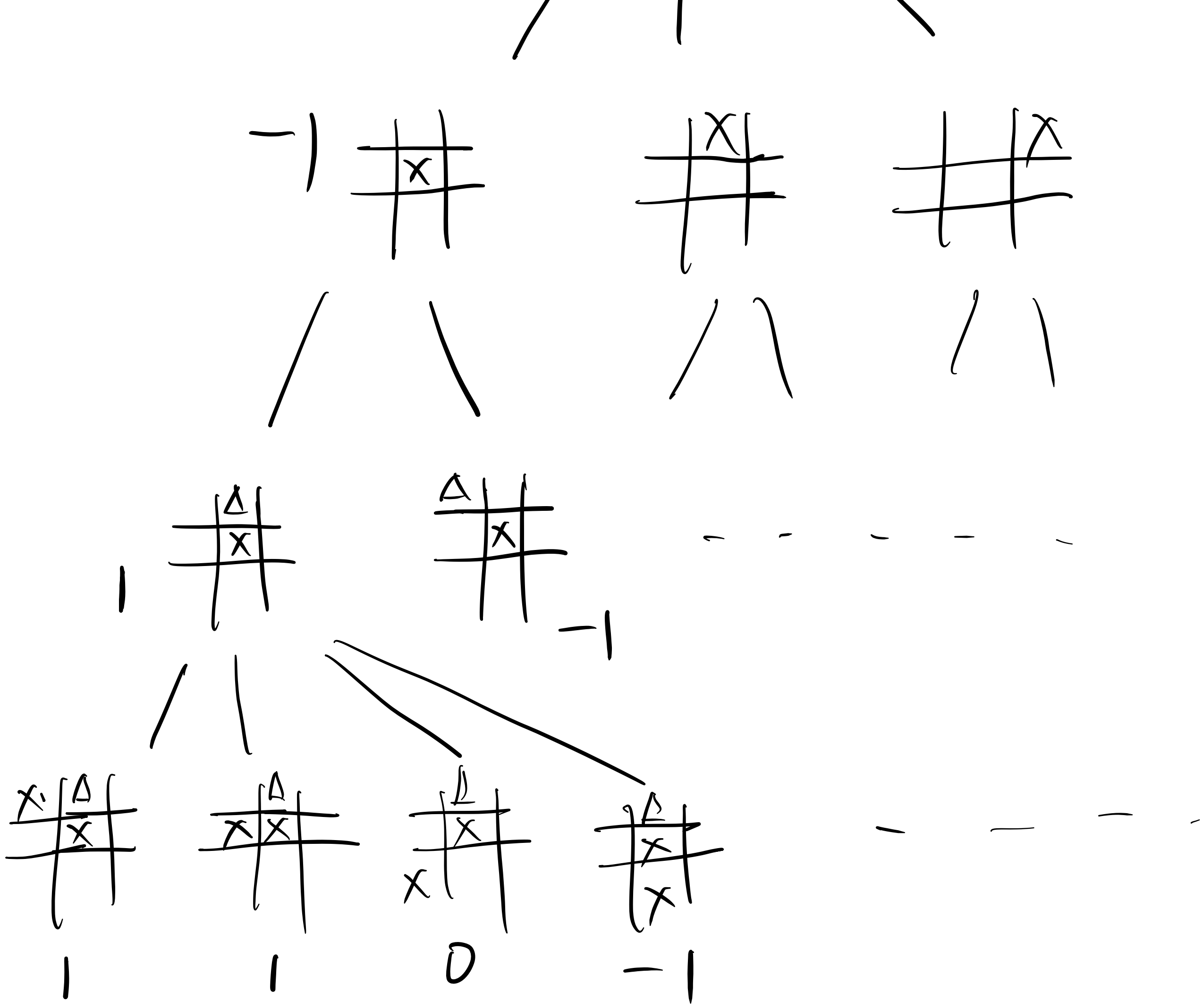
$\Theta(2^n \cdot n \lg n)$ for rare instances

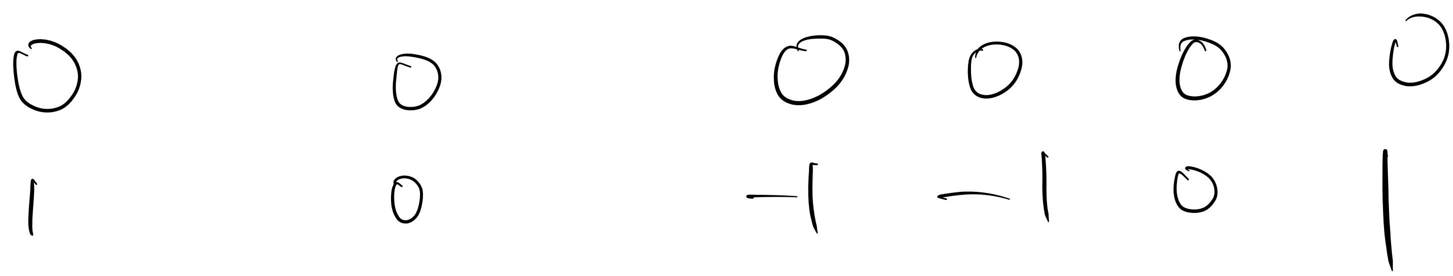
$O(n^2 \cdot \lg n)$ for most instances.

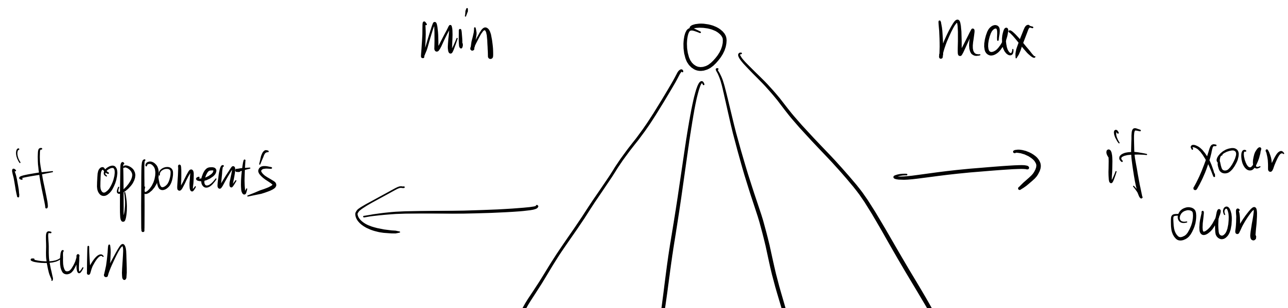
Game

Tic-tac-toe




-1
1
-1
1
1
0
-1

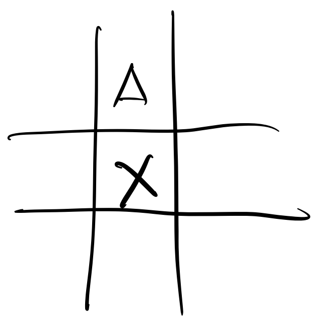

1
0
-1
-1
0
1


min
max
if opponent's turn
if your own

minimax strategy

$$f(P) = W_{you} - W_{opponent}$$

potential wins



$W_{you} = 6$

$W_{opponent} = 4$

$f(P) = 2$