

## Backtracking

1] Basics

1. If  $N$  is a goal node, return "success"

2. If  $N$  is a leaf node, return "failure"

3. For each child  $C$  of  $N$ ;

    explore  $C$

    if  $C$  is successful, return "success"

return "failure"

2] depth first search

\* N queens problem

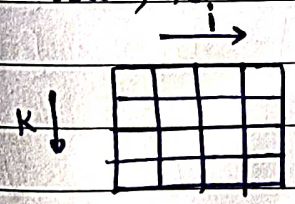
place  $n$  queens in such a manner on an  $n \times n$  chessboard

that no two queens attack each other

Explanation: -  $(x_1 - x_2)^2 + (y_1 - y_2)^2 = (x_1 - x_2)^2$

1] we keep global array  $x[...]$  as final answer in  $n$ -tuple

index represents  $k^{th}$  queen and value at that index represents column on chessboard. Note that  $k^{th}$  queen has to be placed on  $k^{th}$  row, i.e. each new queen in new row



2] for each row, we travel for each column,

3] we check if that cell is available by place( $k, i$ )

4] if available, note that place for  $k^{th}$  queen

by  $x[k] = i$

5] also if we reach to final queen, print  $x$  else travel for next position of next queen

For place( $k, i$ )

1] from initial condition that each queen has to be placed on new row, we make sure, no horizontal check required

2] for vertical check, place( $k, i$ )

for  $j$  from 0 to  $k-1$

$x[j] == i \rightarrow$  if true then return false

included

as this cell not to be taken



3) diagonal check

	0	1	2	3
0				
1		X		
2				
3				

place(k, i)

for j in (0, k-1)

for each queen

left diagonal  $\rightarrow$  row - col stays same

for previous queens  $\rightarrow j - x[j]$

for our current queen which is at (k, i)

equating both  $j - x[j] = k - i$

$j - k = x[j] - i$  — ① for left diagonal

Right diagonal  $\rightarrow$  row + col stays same

previous queens  $\rightarrow j + x[j]$

current queen at (k, i)  $\rightarrow k + i$

equating both  $j + x[j] = k + i$

$j + k = i + x[j]$  — ②

from ① & ②

$abs(j - k) = abs(i - x[j])$  — ③

if this comes true then that space is not available

hence for  $x[j] = i$  & ③ either of them must not be true

if any of these conditions are true then that space is not available

for our current queen



$$x = [-1]^n n$$

# for place holder.

## Assuming array index starting from 0

~~def~~ Place (k,i)

for  $j \leftarrow 0$  to  $k-1$

# for each previous queen

do if  $(x[j] = i)$  or

$$\text{abs}(x[j] - i) = \text{abs}(j - k)$$

```

then return false;

```

return true;

3

### nQueens (k, n)

٩

for  $i \leftarrow 0$  to  $n-1$

# for each col<sup>n</sup>

do if place  $(k, i)$  then

$$\{ \quad x[k] \leftarrow i; \quad$$

# place available so write

~~if  $(k = n)$  then~~

if  $(k = n-1)$  then

# 1<sup>st</sup> queen denoted by 0  
hence  $n-1$

hence  $n-1$

write  $(x[1..n])$

else

inQueens (K+1, n);

# when not reached

last queen

2

3

11/10/1960