How many ways are there to place $n$ queens on an $n \times n$ board, if queens are already on first $r$ rows at positions $Q[1..r]$?
PLACEQUEENS(Q[1..n], r):
  if r = n + 1
    print Q[1..n]
  else
    for j ← 1 to n
      legal ← TRUE
      for i ← 1 to r − 1
        if (Q[i] = j) or (Q[i] = j + r − i) or (Q[i] = j − r + i)
          legal ← FALSE
      if legal
        Q[r] ← j
        PLACEQUEENS(Q[1..n], r + 1)  ⟨(Recursion!)⟩

Figure 2.2. Gauss and Laquière's backtracking algorithm for the n queens problem.
n-queens completion is NP-hard

top-down n-queens completion

OPEN
Figure 2.4. Vera wins the $3 \times 3$ fake-sugar-packet game.
Game state = positions of all pieces

\[ \text{not full history} \]

```
PLAYANYGAME(X, player):
    if player has already won in state X
        return GOOD
    if player has already lost in state X
        return BAD
    for all legal moves \( X \rightsquigarrow Y \)
        if PLAYANYGAME(Y, \neg player) = BAD
            return GOOD \[X \rightsquigarrow Y \text{ is a good move}\]
    return BAD \[(There are no good moves)\]
```
Given a string $A[1...n]$, is $A$ the concat of words?

$\text{Is Word}(w) \iff w$ is a word

$\text{Is Word}(w) \iff \text{True}$ if $w$ is a word

$\text{Is Word}(w) \iff \text{False}$ if $w$ is not a word

- BLUE STEM UNIT ROBOT HEART HANDS ATURN SPIN

- BLUE ST EMU NITRO BOT HEART HANDS ATURN SPIN

Is the suffix $A[i...n]$ the concat of words?
**SPLITTABLE**(A[1..n]):

if \( n = 0 \)
   return \( \text{TRUE} \)

for \( i \leftarrow 1 \) to \( n \)
   if Is\text{WORD}(A[1..i])
      if SPLITTABLE(A[i + 1..n])
         return \( \text{TRUE} \)

return \( \text{FALSE} \)

\[
S\text{plittable}(i) = \begin{cases} 
\text{TRUE} & \text{if } i > n \\
\bigvee_{j=i}^{n} (\text{Is\text{WORD}}(i, j) \land \text{SPLITTABLE}(j + 1)) & \text{otherwise}
\end{cases}
\]

**SPLITTABLE**(i):

if \( i > n \)
   return \( \text{TRUE} \)

for \( j \leftarrow i \) to \( n \)
   if Is\text{WORD}(i, j)
      if SPLITTABLE(j + 1)
         return \( \text{TRUE} \)

return \( \text{FALSE} \)

But only \( n \) different ways to call this function

Write down results! \( \Rightarrow O(n^2) \) time (calls to Is\text{WORD})