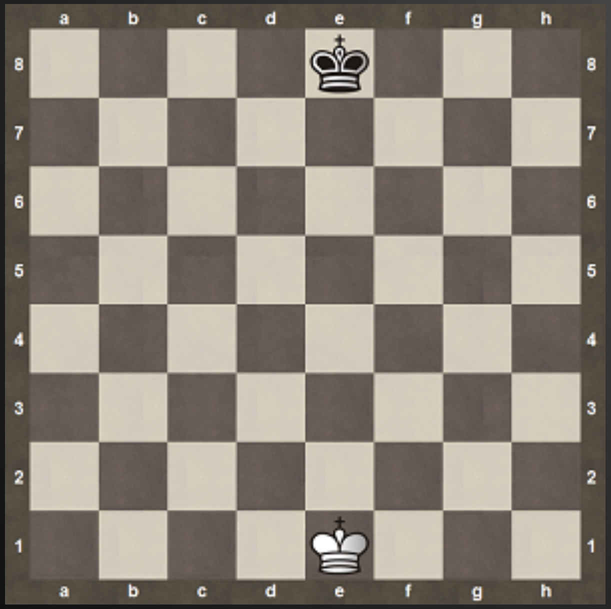
Advanced Algorithmic and Programming Techniques

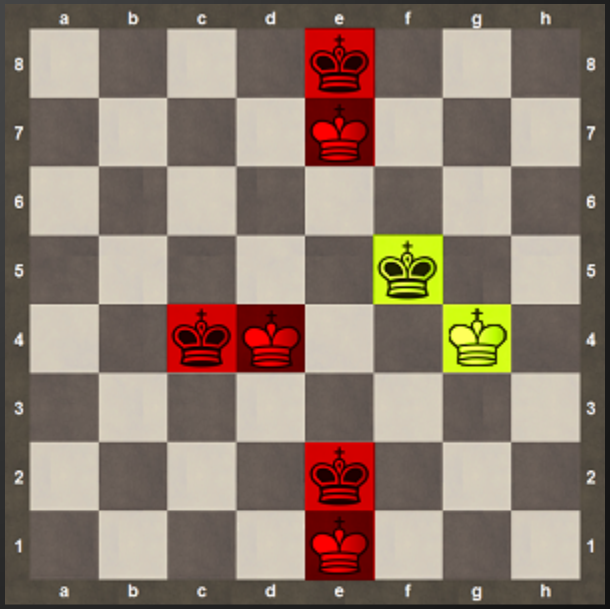
Motivational Problem

Lecturers: Marko Genyk-Berezovskyj, Tomáš Tunys

Problem: Kings on the Chessboard

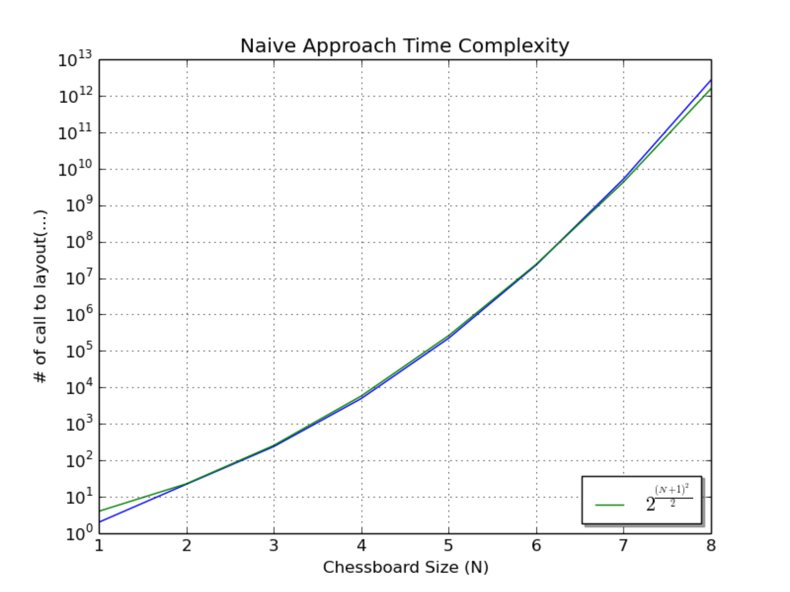
- In how many ways can you lay out *4-way* kings on an 8x8 chessboard such that no two kings threaten each other?





Solution #1: Brute Force

- Do precisely what the problem assignment says, i.e. try every possible lay out of the kings and check whether there is no pair of kings threatening each other.



Solution #2: Exploit Column Layouts

- Consider every possible layout of kings in a column. (How many are there?)

Solution #2: Exploit Column Layouts

- Consider every possible layout of kings in a column. (How many are there?)

Column Size	Possible Layouts
1	2
2	3
3	5
4	8
5	13
6	21
7	34
8	**55**

Solution #2: Exploit Column Layouts

- Consider every possible layout of kings in a column. (How many are there?)

Column Size	Possible Layouts
1	2
2	3
3	5
4	8
5	13
6	21
7	34
8	**55**

Fibonacci Sequence!

Solution #2: Exploit Column Layouts

- Let $L(n, c)$ denote the number of layouts up to column n with configuration c. Can you find the answer for $L(n+1, c)$ using $L(n, c')$?

Solution #2: Exploit Column Layouts

- Let $L(n, c)$ denote the number of layouts up to column n with configuration c. Can you find the answer for $L(n+1, c)$ using $L(n, c')$?
- $L(n + 1, c')$ is sum over all $L(n, c)$ for which c and c' can be laid out beside each other.
- Base case: $L(1, c) = 1$ for every c.
- **Answer:** sum of $L(N, c)$ for every c.

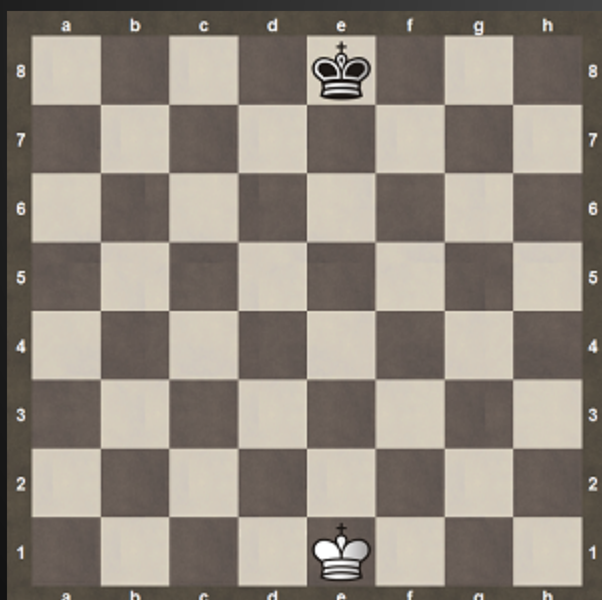
Solution #2: Exploit Column Layouts

- Time complexity: **$O(N * \#C^2)$**, where #C is the number of column layouts.

```
for (n = 1; n < BOARD_SIZE; n++)
{
    for (c = 0; c < COLUMN_LAYOUTS; c++)
    {
        L[n][c] = 0;
        for (cc = 0; cc < COLUMN_LAYOUTS; cc++)
        {
            if (D[c][cc])
            {
                L[n][c] = (L[n][c] + L[n - 1][cc]) % MOD;
            }
        }
    }
}
```

Problem: Kings on the Chessboard

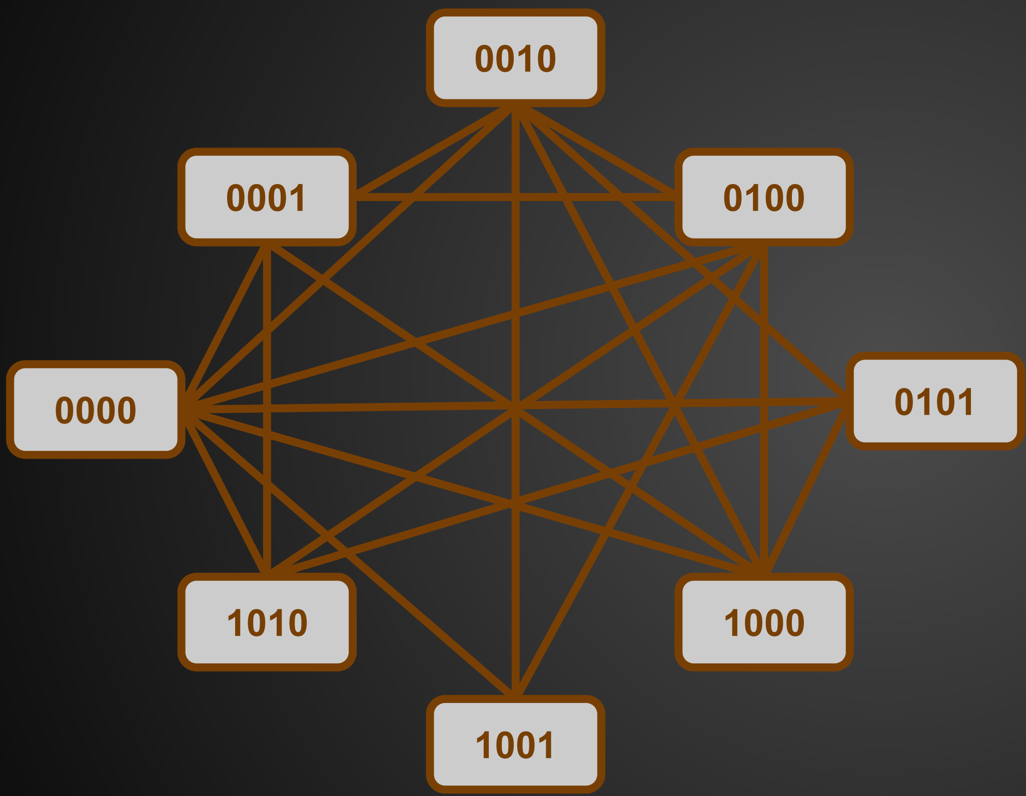
- In how many ways can you lay out *4-way* kings on an 8xN chessboard such that no two kings threaten each other?



We know that the problem’s time complexity grows exponentially with the size of the chessboard, but what if we fix the height, but let its width (N) grow...

… can you find the answer for N as large as 10^{18}?

Solution #3: Matrix Exponentiation



A

0	1	1	1	1	1	1	1
1	0	1	1	0	1	0	1
1	1	0	1	1	1	1	0
1	1	1	0	0	1	1	1
1	0	1	0	0	0	0	1
1	1	1	1	0	0	0	0
1	0	1	1	0	0	0	0
1	1	0	1	1	0	0	0

$A[C][C'] = 1$ iff $C \,\&\, C' = 0$

Solution #3: Matrix Exponentiation

- **Theorem:** Let A be an adjacency matrix of graph G, then $(A^k)[i][j]$ is the number of distinct sequences of k edges connecting vertex i with vertex j. $A^0 = I$ (identity matrix)
- **Proof:** by induction.

A

0	1	1	1	1	1	1	1
1	0	1	1	0	1	0	1
1	1	0	1	1	1	1	0
1	1	1	0	0	1	1	1
1	0	1	0	0	0	0	1
1	1	1	1	0	0	0	0
1	0	1	1	0	0	0	0
1	1	0	1	1	0	0	0

A[C][C’] = 1 iff C & C’ = 0

Solution #3: Matrix Exponentiation

- A^k can be computed in $O(N^3 \log(k))$ using *exponentiation by squaring* (N is the size of the matrix A).
- How to get the final answer from the matrix A^N?
- Time complexity:

  $O(\#C^3 \log(N))$

A

0	1	1	1	1	1	1	1
1	0	1	1	0	1	0	1
1	1	0	1	1	1	1	0
1	1	1	0	0	1	1	1
1	0	1	0	0	0	0	1
1	1	1	1	0	0	0	0
1	0	1	1	0	0	0	0
1	1	0	1	1	0	0	0

A[C][C'] = 1 iff C & C' = 0

Answers

- Number of layouts for the 8x8 chessboard (modulo $10^9 + 7$):

647958335

- Number of layouts for the 8×10^{18} chessboard (modulo $10^9 + 7$):

795080988