NP-Complete Problems
(Algorithmic Problems)

NP-Complete Problems

- we can reduce the original **O(N!)** brute-force approach to **exponential running time** with backtracking for the **queens problem**
- the **coloring problem** can have **exponential running time** with backtracking
- **BUT EXPONENTIAL RUNNING TIME IS VERY SLOW**
- we can not use algorithms with exponential running time complexities in real-world applications
- usually **NP-complete** and **NP-hard** problems have extremely high number of possible states – so the algorithms will be extremely slow

Solution: Meta-Heuristics and AI

- Most of the problems are **NP-complete** or **NP-hard**
- then how to deal with these kinds of problems?
- we can use **artificial intelligence** and **meta-heuristics**
- the idea is that let's not find the exact solution because it takes a lot of time because of the exponential running time of algorithms
- we are after an **approximation**
- these approaches will not find the optimal solution but at least they are fast – we do not have to wait years for the results
- ***genetics algorithms***, ***simulated annealing***, ***ant colony optimization***