Constraint Satisfaction Problems

CZ3005: Artificial Intelligence

Shen Zhiqi

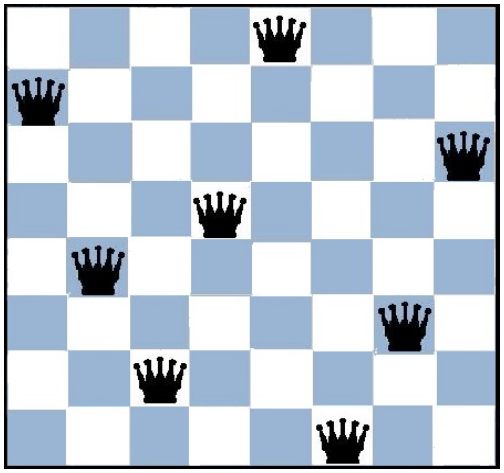
Outline

- What is the Constraint Satisfaction Problem?
- Backtracking search
- Forward checking and constraint propagation
- Most-constraining variable and least-constraining value heuristics
- Search using min-conflict heuristics

Constraint Satisfaction Problem (CSP)

Goal: discover some state that satisfies a given set of constraints

Example: 8-Queens Problem



Example: Cryptarithmetic Puzzle

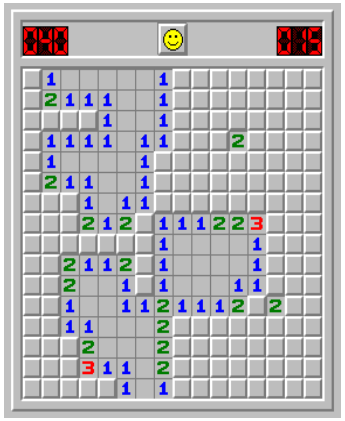
```
   S  E  N  D
+  M  O  R  E
-------------
M  O  N  E  Y
```

Constraint Satisfaction Problem (CSP) cont.

Example: Sudoku

							1	3
			7					6
			5		9			
						9		
1		6						
						2		
7	4						5	
	8					4		
				1				

Example: Minsweeper



Examples: Real-world CSPs

- Assignment problems
  - e.g., who teaches what class
- Timetabling problems
  - e.g., which class is offered when and where?
- Hardware configuration
- Transportation scheduling
- Factory scheduling
- Floor-planning

CSP

State

- defined by variables V_i with values from domain D_i

Example: 8-queens

- Variables: locations of each of the eight queens
- Values: squares on the board

Goal test

- a set of constraints specifying allowable combinations of values for subsets of variables

Example: 8-queens

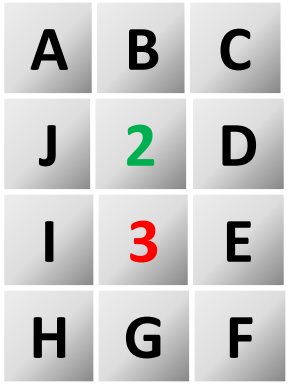
- Goal test: No two queens in the same row, column or diagonal

Example: Cryptarithmetic Puzzle

$$\begin{array}{r} S\ E\ N\ D \\ +\ M\ O\ R\ E \\ \hline M\ O\ N\ E\ Y \end{array}$$

- Variables: D, E, M, N, O, R, S, Y
- Domains: {0, 1, 2, 3, 4, 5, 6, 7, 8, 9}
- Constraints
  - *Y = D + E or Y = D + E - 10*, etc
  - D ≠ E, D ≠ M, D ≠ N, etc.
  - M ≠ 0, S ≠ 0 (unary constraints: concern the value of a single variable)

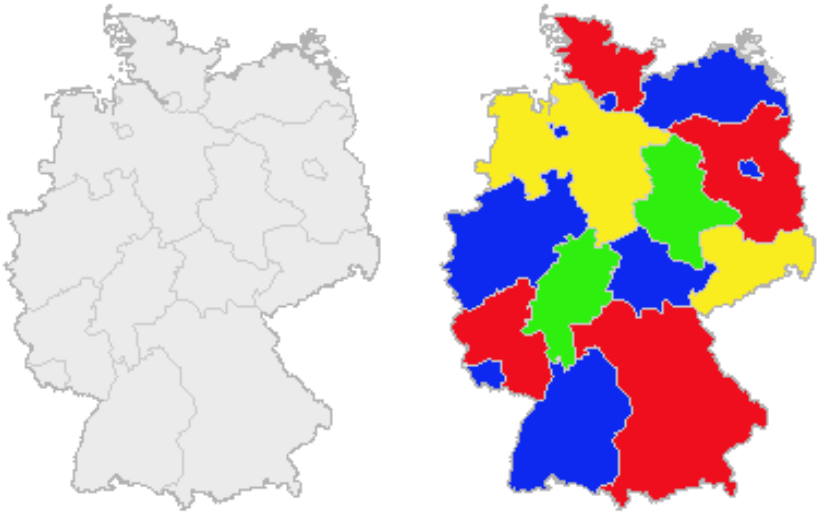
Example: Minesweeper



- Variables: The cells
- Domains: {0; 1} representing {safe, mined}
- Constraints: Each cell has a number $m \in \{ 1, \dots , 8\}$ indicating the number of mines nearby, so m is equal to sum of value of neighbour cells

Example: Map Coloring

Color a map so that no adjacent parts have the same color



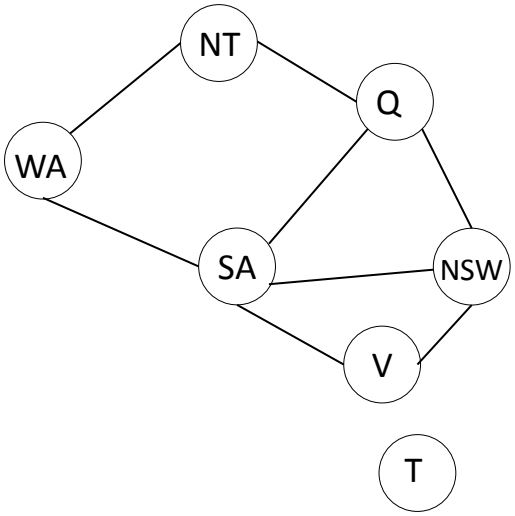
- Variables: Countries *Ci*
- Domains: {*Red, Blue, Green*}
- Contraints: C1 ≠ C2, C1 ≠ C5, etc
  - binary constraints

Some Definitions

- A state of the problem is defined by an **assignment** of values to some or all of the variables
- An assignment that does not violate any constraints is called a **consistent** or **legal** assignment
- A **solution** to a CSP is an assignment with every variable given a value (**complete**) and the assignment satisfies all the constraints

Visualize a CSP as a Constraint Graph

Example: Australia map coloring



Nodes: Variables; Arcs/Edges: Constraints.

Applying Standard Search

- States: defined by the values assigned so far
- Initial state: all variables unassigned
- Actions: assign a value to an unassigned variable
- Goal test: all variables assigned, no constraints violated

Question: How to represent constraints?

Answer: Explicitly (e.g., D ≠ E)

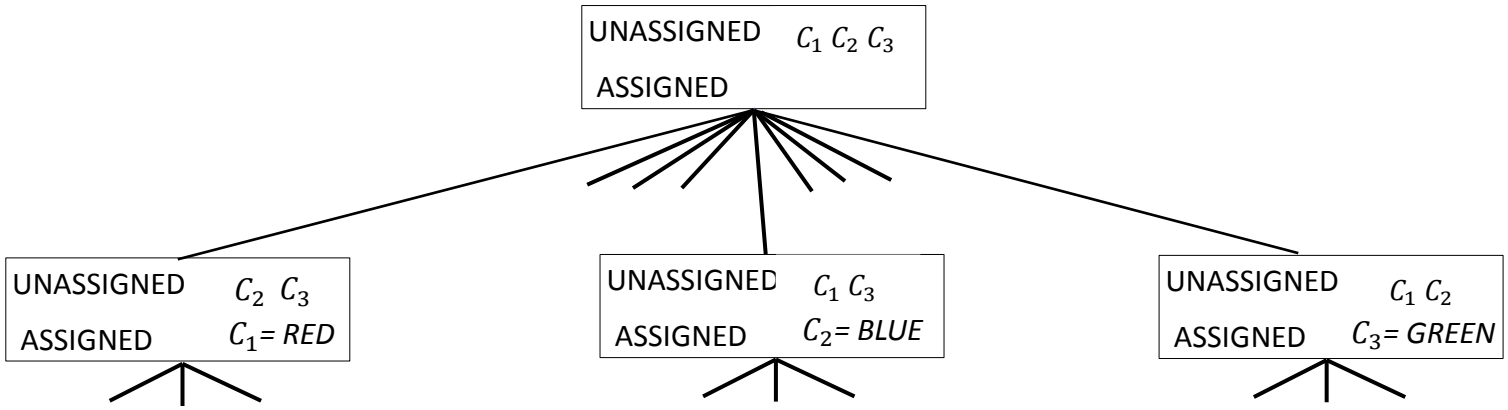
Example

- Row the 1st queen occupies: $V_1 \in \{1, 2, 3, 4, 5, 6, 7, 8\}$
  (similarly, for V_2)
- No-attack constraint for V_1 and V_2:
  { <1, 3> , <1, 4>, <1, 5>, … , <2, 4>, <2, 5>, …}

Implicitly: use a function to test for constraint satisfaction

Applying Standard Search...

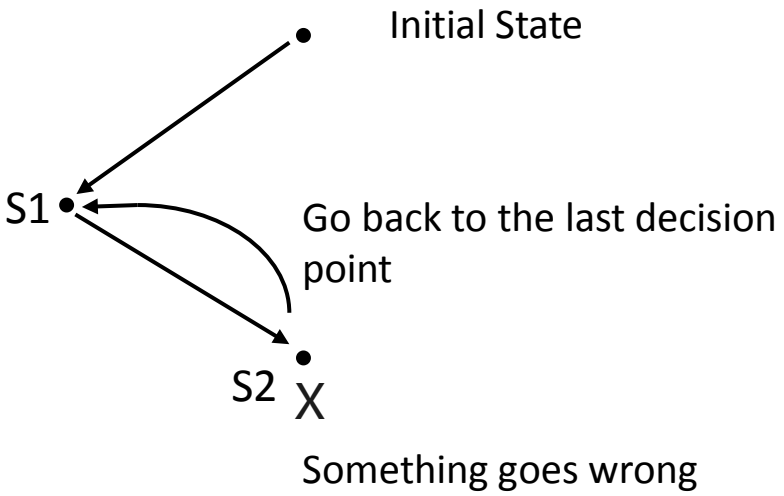
Example: map coloring



- Number of variables: n
- Max. depth of space: n
- Depth of solution state: n (all variables assigned)
- Search algorithm: depth-first search

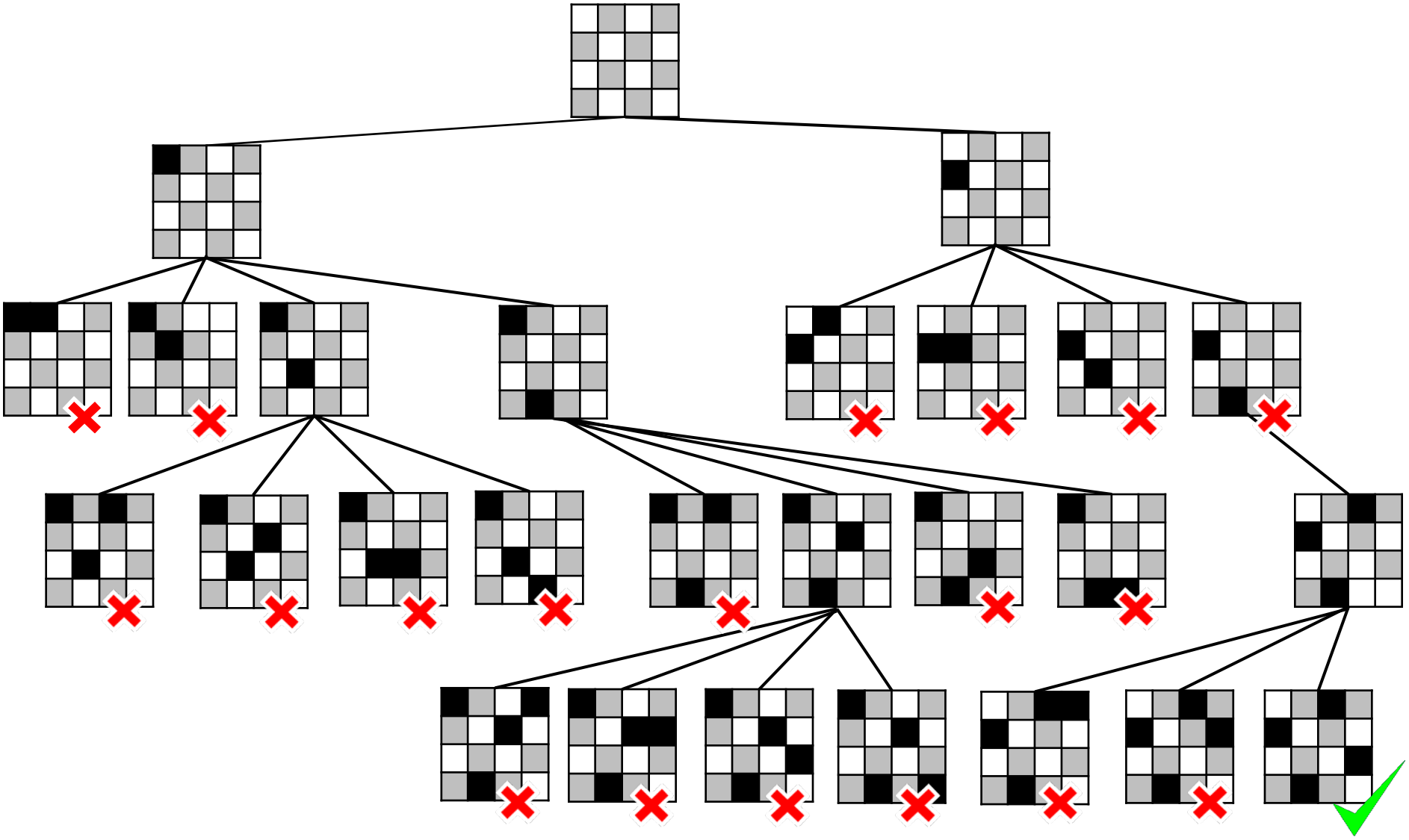
Backtracking Search

Backtracking search: Do not waste time searching when constraints have already been violated



- Before generating successors, check for constraint violations
- If yes, backtrack to try something else

Example (4-Queens)



Heuristics for CSPs

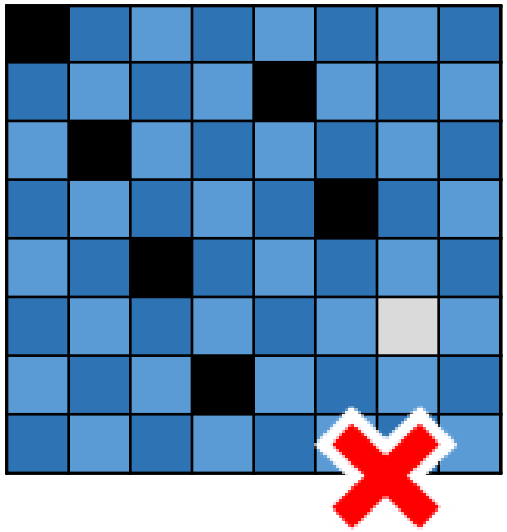
Plain backtracking is an uninformed algorithm!!

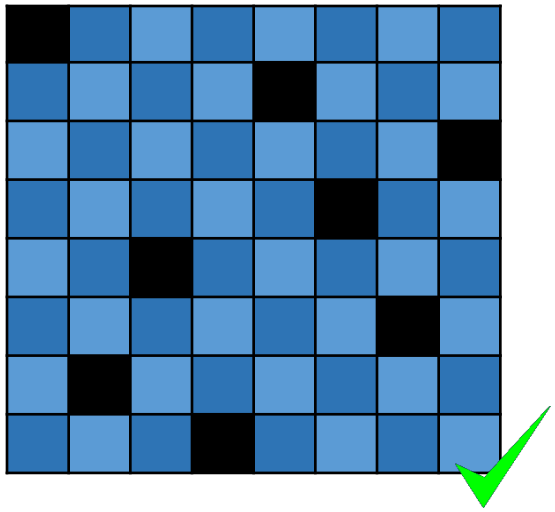
More intelligent search that takes into consideration

- Which variable to assign next
- What order of the values to try for each variable
- Implications of current variable assignments for the other unassigned variables
  - forward checking and **constraint propagation**

Constraint propagation: propagating the implications of a constraint on one variable onto other variables

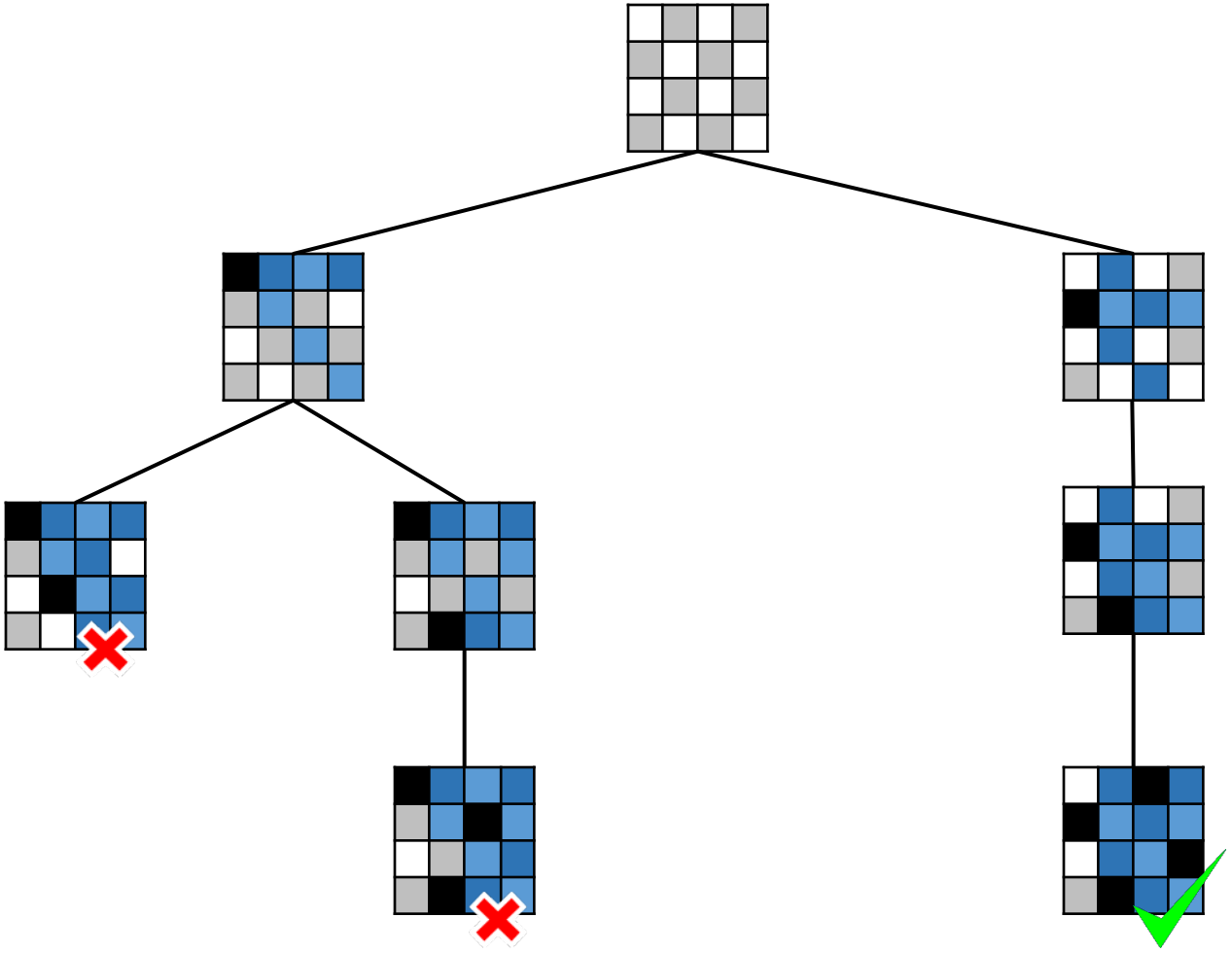
Constraint Propagation for 8-Queens



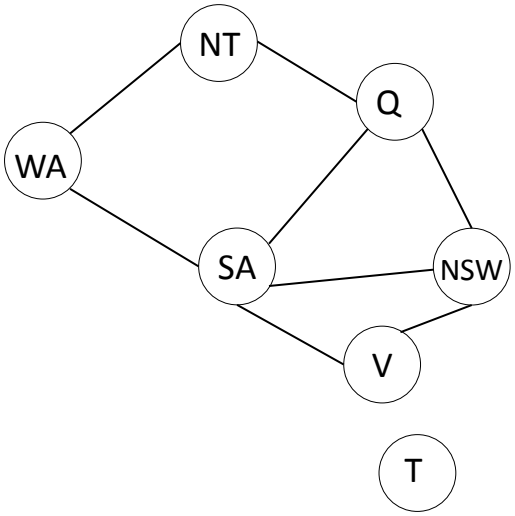


Solution for 8-Queens

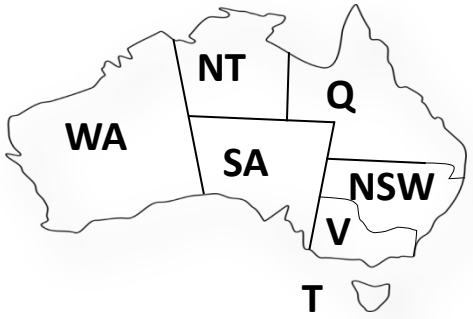
Search Tree of 4-Queens with Constraint Propagation



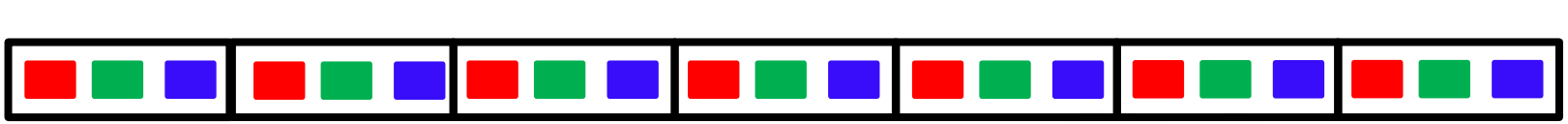
Example (Map Coloring)

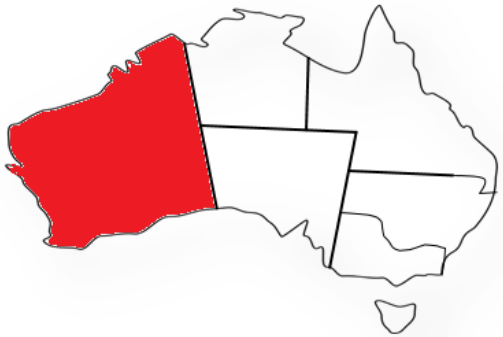


Example (Map Coloring)…

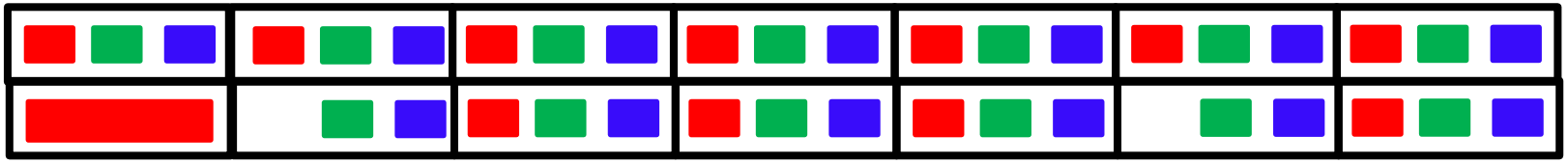


WA NT Q NSW V SA T



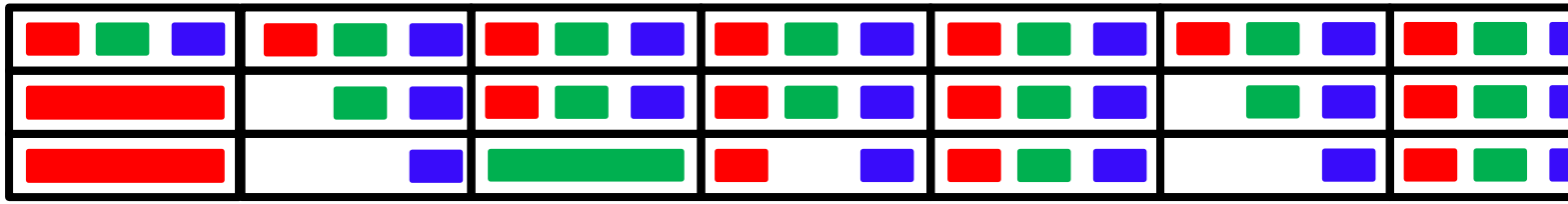


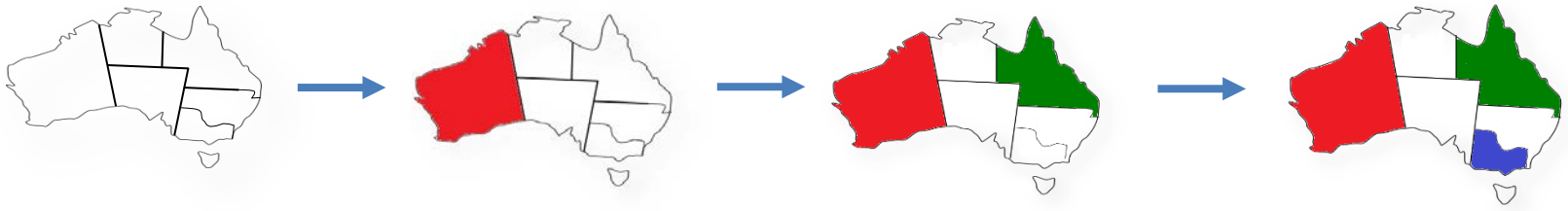
WA NT Q NSW V SA T



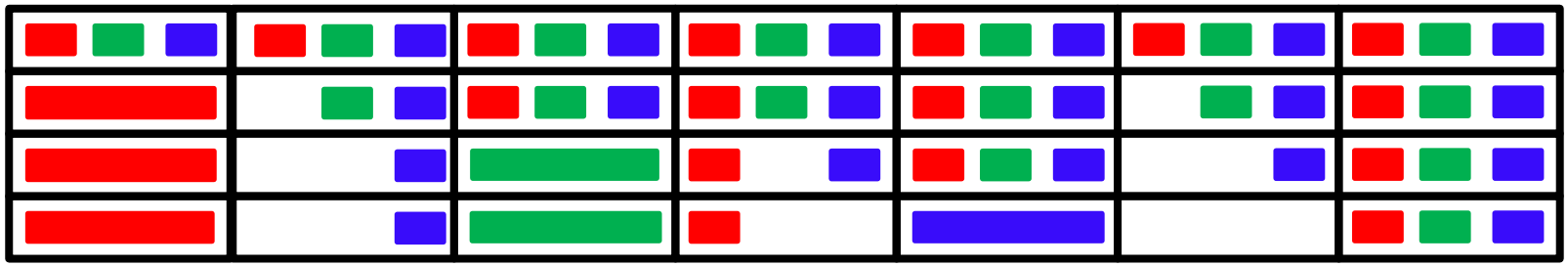
Example (Map Coloring)...

WA NT Q NSW V SA T



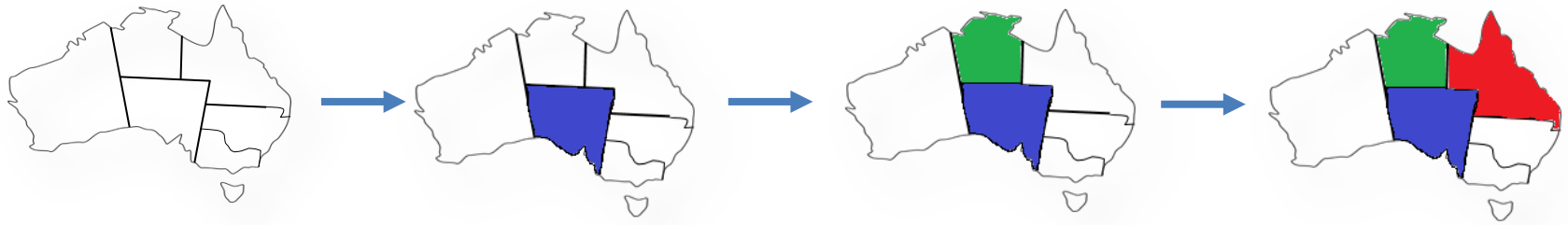


WA NT Q NSW V SA T



Most Constrained Variable (or minimum remaining values (MRV) heuristic

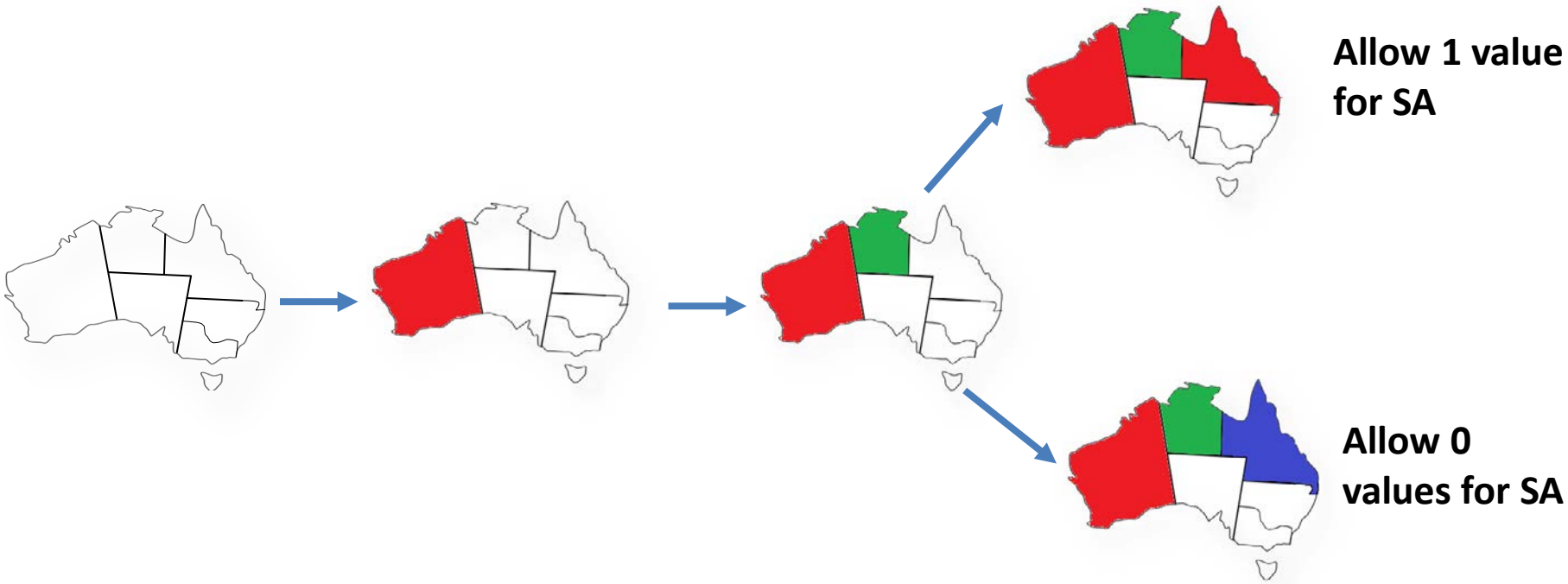
Example: map coloring



To reduce the branching factor on future choices by selecting the variable that is involved in the **largest number of constraints** on unassigned variables.

Not covered in any exam

Least Constraining Value

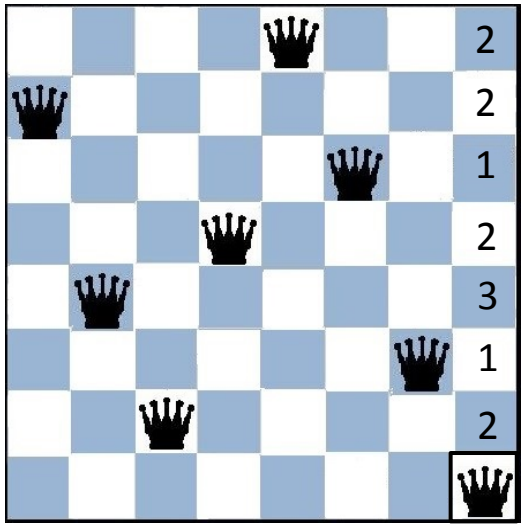


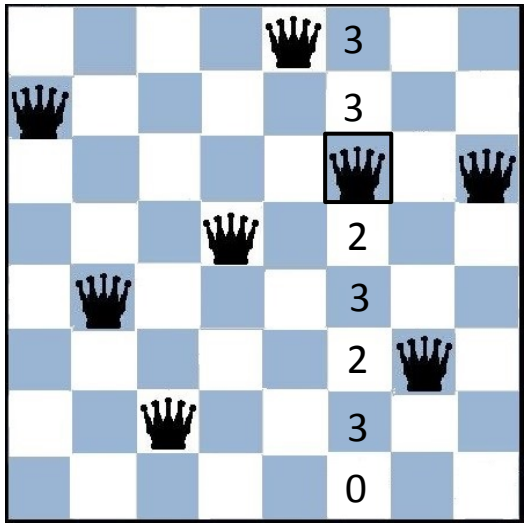
Choose the value that leaves maximum flexibility for subsequent variable assignments

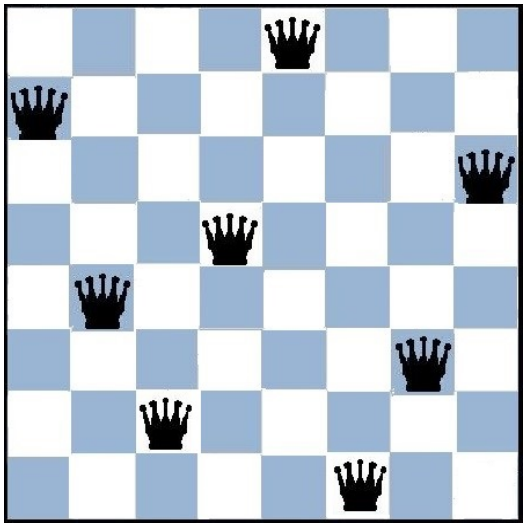
Not covered in any exam

Min-Conflicts Heuristic (8-queens)

- A local heuristic search method for solving CSPs
- Given an initial assignment, selects a variable in the scope of a violated constraint and assigns it to the value that minimizes the number of violated constraints







Not covered in any exam