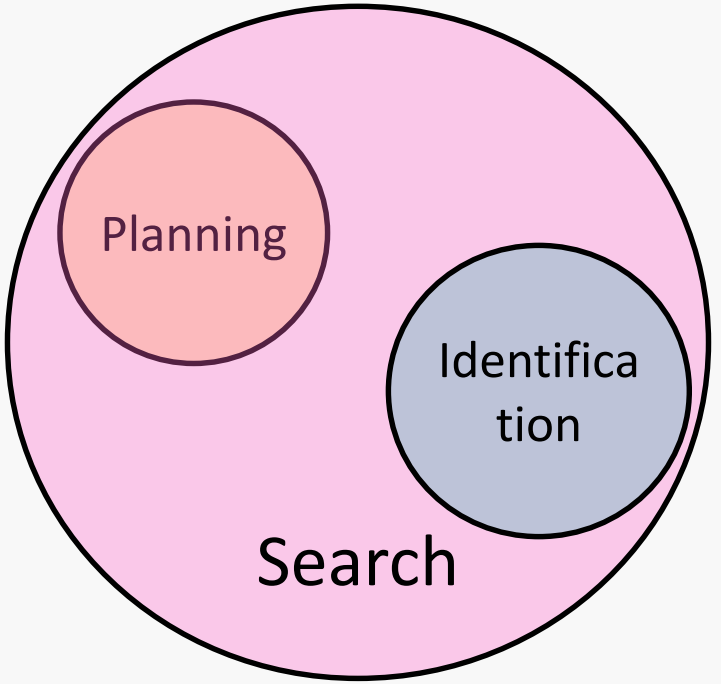
Constraint Satisfaction Problems

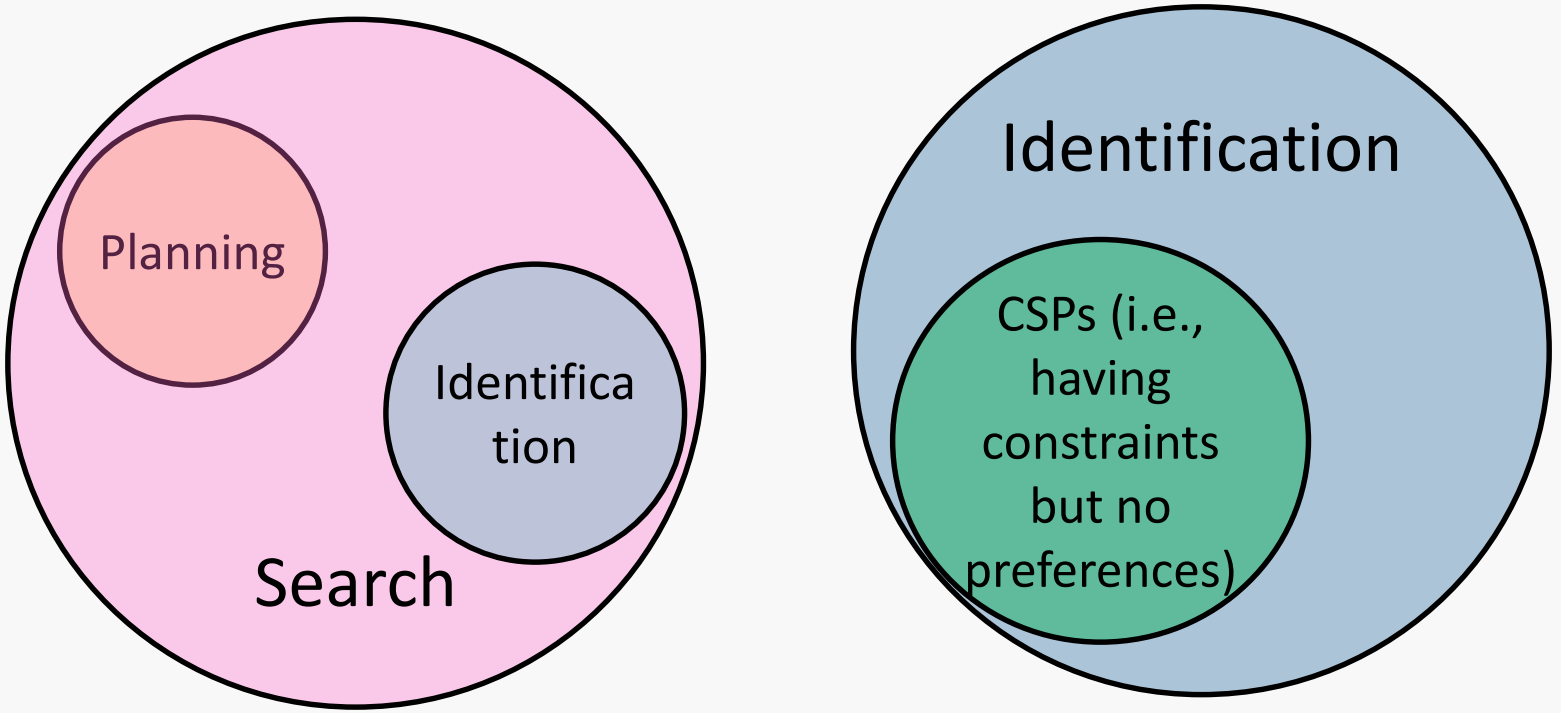
Another type of search problems: Identification

- All the above cases are about planning, which is only one type of search problems.



- Planning: a sequence of actions.
  - We care about the path to the goal.
- Identification: an assignment
  - We care about the goal itself, not the path.
  - For example, a taxi firm assigns taxis a, b, c to customers x, y, z such that the cost incurred (e.g., fuel) is minimal.

Search -> Identification -> CSP



- Constraint Satisfaction Problems (CSPs): Identification problems have constraints to be satisfied; there is no preference in CSPs.
- Constraints refer to hard constraints which a legal solution cannot violate.
- Preferences sometimes are referred to as soft constraints (or objectives), where we need to optimise, e.g., to minimise cost.

CSPs

- A constraint satisfaction problem consists of
  - A set of variables
  - A domain for each variable
  - A set of constraints
- In a CSP, an assignment is complete if every variable has a value, otherwise it is partial. Solutions are complete assignments satisfying all the constraints.
- Example: Module scheduling problem
  - Variables – Modules: AI1, Data Structure, Software Engineering, OOP, AI2, Neural Computation, Evolutionary Computation…
  - Domain - year-term: {1-1, 1-2, 2-1, 2-2, 3-1, 3-2}
  - Constraints - pre-requisites: {AI1 < AI2, OOP < SE…}
  - Solutions - e.g.: (AI1=1-2, OOP=1-1, AI2=2-2, SE=2-1…)

Standard Search Problems vs CSPs

- Standard Search problems
  - State is a “black-box”: arbitrary data structure
  - Goal test can be any function over states

- Constraint Satisfaction Problems (CSPs)
  - State is defined by variables X_1, X_2,… with values from domains D_1, D_2, …
  - Goal test is a set of constraints specifying allowable combinations of values of variables.
  - An example of a formal representation language, in which many search problems can be abstracted. This allows useful general-purpose algorithms with more power than standard search algorithms.

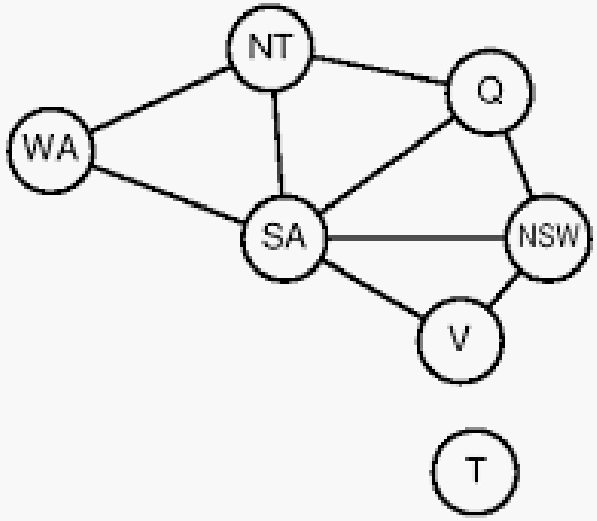
Example: Map Colouring for Australia

Problem: Map colouring problem is to paint a map (e.g., via three colours red, green and blue) in such a way that none of adjacent regions can have the same colour.

- Variables: WA, NT, Q, NSW, V, SA, T
- Domain: D = {red, green, blue}
- Constraints: adjacent regions must have different colours
  - WA ≠ NT, WA ≠ SA, NT ≠ SA, NT ≠ Q,…
- Solutions: e.g. {WA=red, NT=green, Q=red, NSW=green, V=red, SA=blue, T=green}

Constraint Graphs

- Constraint graphs are used to represent relations among constraints in CSPs, where nodes correspond to the variables and arcs reflect the constraints.



What is the difference between a constraint graph and a search state graph?

Example: Einstein Puzzle

Problem: There are two houses, and each house has a different colour (either blue or red), and a different pet (cat, dog or fish). We also have the following constraints: The first house does not have a dog nor fish. The blue house has a fish. Which colour and pet each house has?

- Variables: Colour1, Colour2, Pet1, Pet2
- Domain:
  - Colour1, Colour2 = {blue, red}
  - Pet1, Pet2 = {cat, dog, fish}
- Constraints:
  - Pet1 ≠ dog, fish
  - If Colour1 = blue, then Pet1 = fish
  - If Colour2 = blue, then Pet2 = fish
  - Colour1 ≠ Colour2, Pet1 ≠ Pet2

Example: Sudoku

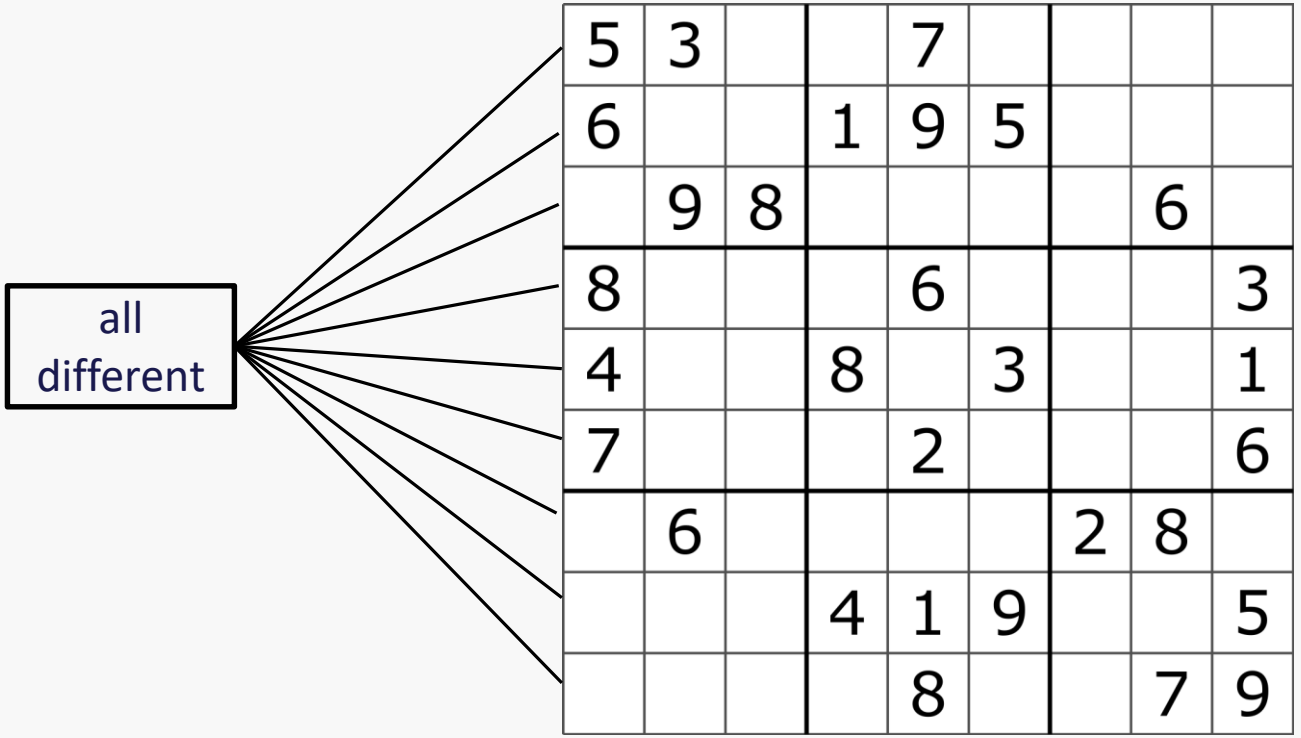
Problem: Sudoku is to fill a 9×9 grid with digits so that each column, each row, and each of the regions contain all of the digits from 1 to 9.

- Variables: each open cell
- Domain: D = {1,2,3,…,9}
- Constraints:
  - Each row contains different numbers
  - Each column contains different numbers
  - Each region contains different numbers

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

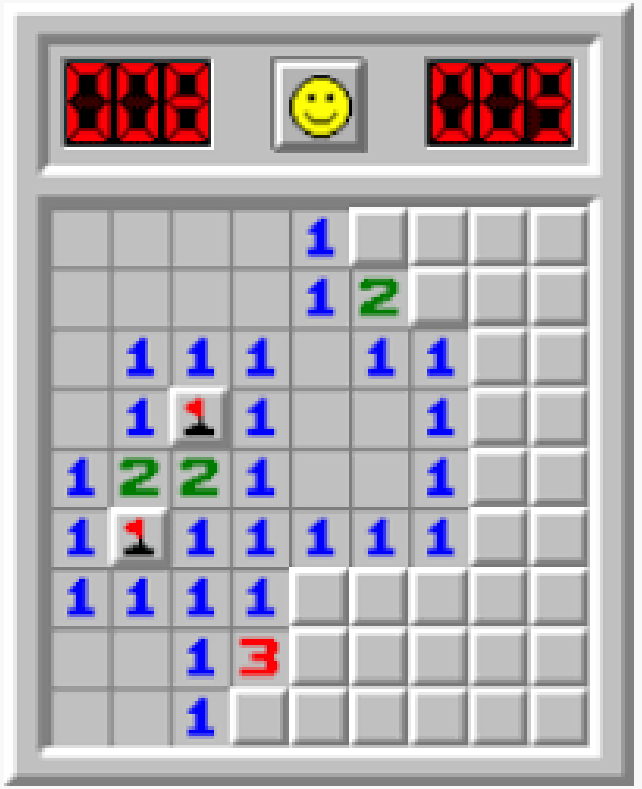
How to draw a constraint graph when a constraint relates to more than two variables?

- Use a square to represent a constraint, and connect all the variables involved in that constraint.



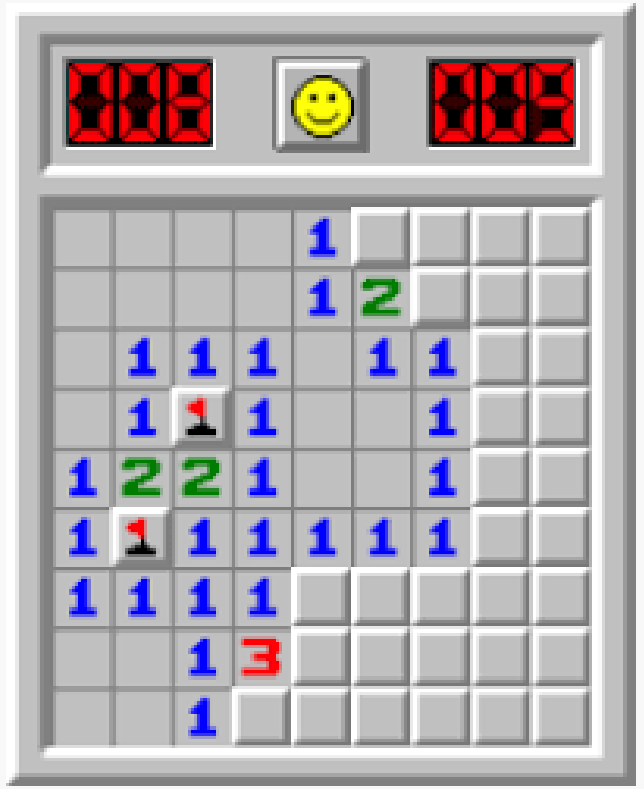
Minesweeper

- Minesweeper is a single-player puzzle game. The goal is to not uncover a square that contains a mine; if you've identified a square that you think it is a mine, then you flag it.
- When you uncover a square, if the square is a mine, the game ends and you lost.
- Otherwise, it is a numbered square indicating how many mines are around it (between 0 and 8). If you uncover all of the squares except for any mines, you win the game.



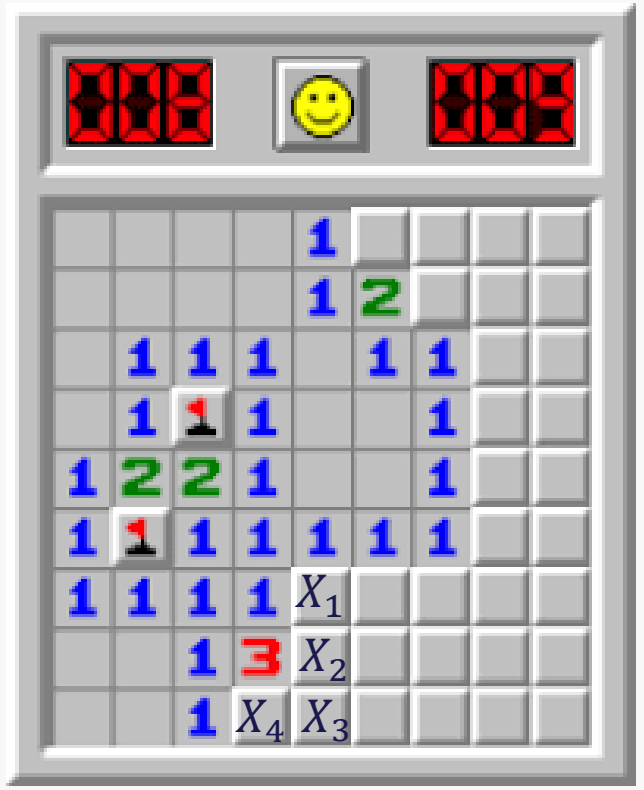
Formalise Minesweeper as a CSP

- Variables:
  - All squares to be uncovered X_1, X_2,…
- Domain:
  - $D = \{0, 1\}$, where 0 denotes not a mine and 1 denotes a mine
- Constraints:
  - The number on a square is the sum of its neighbour's values.



Minesweeper - example

- Variables:
  - X_1, X_2, X_3, X_4
- Domain:
  - $D = \{0, 1\}$, where 0 denotes not a mine and 1 denotes a mine.
- Constraints:
  - $X_1 = 1$
  - $X_1 + X_2 = 1$
  - $X_1 + X_2 + X_3 + X_4 = 3$
  - $X_4 = 1$
  - …



Variety of CSPs

- Variables
  - Finite domains (discrete), e.g. all the preceding examples.
  - Infinite domains (discrete or continuous), e.g., variables involving time.
- Constraints
  - Unary, binary and high-order constraints; i.e., how many variables involved in a constraint.
- CSPs are difficult search problems
  - If a CSP has n variables, the size of each domain is d, then there are $O(d^n)$ complete assignments.
  - For the preceding representation of the 4 × 4 queens puzzle, there are 2^{16} complete assignments.

Real-world CSPs

- Assignment problems, e.g. who teaches which class
- Timetabling problems, e.g. which class is offered when and where
- Hardware configuration
- Transportation scheduling
- Factory scheduling
- Circuit layout
- Fault diagnosis
- …

Many of CSP problems can also consider the preferences (i.e., objectives), in which case they turn into constrained optimisation problems.

Homework: Cryptarithmetic

Cryptarithmetic is a puzzle where the digits of numbers are represented by letters. Each letter represents a unique digit. The goal is to find the digits such that a given equation is verified.

Question: Formalise the cryptarithmetic problem as a CSP, i.e., formally give its variables, domain and constraints.

```
    T W O          8 3 6
+   T W O      +   8 3 6
-----------    -----------
  F O U R        1 6 7 2
```