Contest

Jugs Problem

- Classic *AI planning* problem
- Planning: Find a sequence of actions to achieve some goal
- Goal: Target to be achieved
- Actions:
  - Empty a jug
  - Pour from one to another
  - Fill up a jug.

State Space Representation

Capacities: 4 7 13

Starting State

	Jug1	Jug2	Jug3
Water	0	0	0

Action: Fill Up Jug 1

Jug1	Jug2	Jug3
4	0	0

Action: Fill Up Jug 2

Jug1	Jug2	Jug3
0	7	0

Action: Pour from 2 to 1

Jug1	Jug2	Jug3
4	3	0

State Space Representation

Capacities: 4 7 13

(0, 0, 0)

	Jug1	Jug2	Jug3
Water	0	0	0

Action: Fill Up Jug 2

(0, 7, 0)

Jug1	Jug2	Jug3
0	7	0

Action: Pour from 2 to 1

(4, 3, 0)

Jug1	Jug2	Jug3
4	3	0

Searching through State Space

- Use a three-dimensional array, *states[][][]*
  - An entry *states[i][j][k]* maintains:
    - If the state (i, j, k) is reachable so far.
    - The minimum amount of water required to reach it.
  - Initially only (0, 0, 0) is set to be reachable
- Exhaustively fill this array
  - For each currently reachable state,
    - For each possible action (out of 9)
      - Find the new state after completing the action
      - Set the corresponding states[][][] entry to be reachable.
  - 9 * n^3 operations (feasible because n < 25)

Ramsey Theory Problem

- Simple exhaustive algorithm
  - Enumerate all subsets of size *K* in lexicographical order
  - Check if the members of each subset know each other or don’t know each other
  - Output the first answer
    - Since we are enumerating in lexicographical order anyway
- Almost impossible to do it much faster
  - One of the canonical *hard* problems in CS

Ramsey Theory Problem

- Enumerating all subsets in lexicographical order
  - K = 4, N = 10
  - First: (0, 1, 2, 3)
  - Next: (0, 1, 2, 3+1 = 4)
  - …
  - (0, 1, 2, 8+1 = 9)
  - (0, 1, 2, 9+1) is not allowed
  - So, we look for the leftmost number that can be incremented
  - Next: (0, 1, 2+1 = 3, 3+1 = 4)
    - Must reset the numbers to the left to the minimum possible

  - Examples:
    - (1, 3, 8, 9) → (1, 3+1 = 4, 4+1 = 5, 5+1 = 6)
    - (2, 6, 7, 8) → (2, 6, 7, 8+1 = 9)
    - (2, 7, 8, 9) → (2+1 = 3, 4, 5, 6)