NP-Completeness

CptS 223 – Advanced Data Structures

Larry Holder
School of Electrical Engineering and Computer Science
Washington State University

Hard Graph Problems

- "Hard" means no known solutions with better than exponential worst-case running time
- Find a simple path that traverses each edge once (Euler circuit): <u>Easy</u>
- Find a simple path that visits each vertex once (Hamiltonian cycle): <u>Hard</u>
- Single-source, unweighted shortest path problem: <u>Easy</u>
- Longest simple path problem: <u>Hard</u>
- Find simple, shortest path in a weighted graph that visits every vertex once (Traveling Salesman Problem): <u>Hard</u>

Running Times for Hard Problems

Input Size vs. Complexity	10	20	30	40	50	60
n	.00001 s	.00002 s	.00003 s	.00004 s	.00005 s	.00006 s
n^2	.0001 s	.0004 s	.0009 s	.0016 s	.0025 s	.0036 s
n^3	.001 s	.008 s	.027 s	.064 s	.125 s	.216 s
n^5	.1 s	3.2 s	24.3 s	1.7 min	5.2 min	13.0 min
2^n	.001 s	1.0 s	17.9 min	12.7 days	35.7 years	366 centuries
3^n	.059 s	58 min	6.5 years	3855 centuries	2×10^8 centuries	1.3×10^{13} centuries

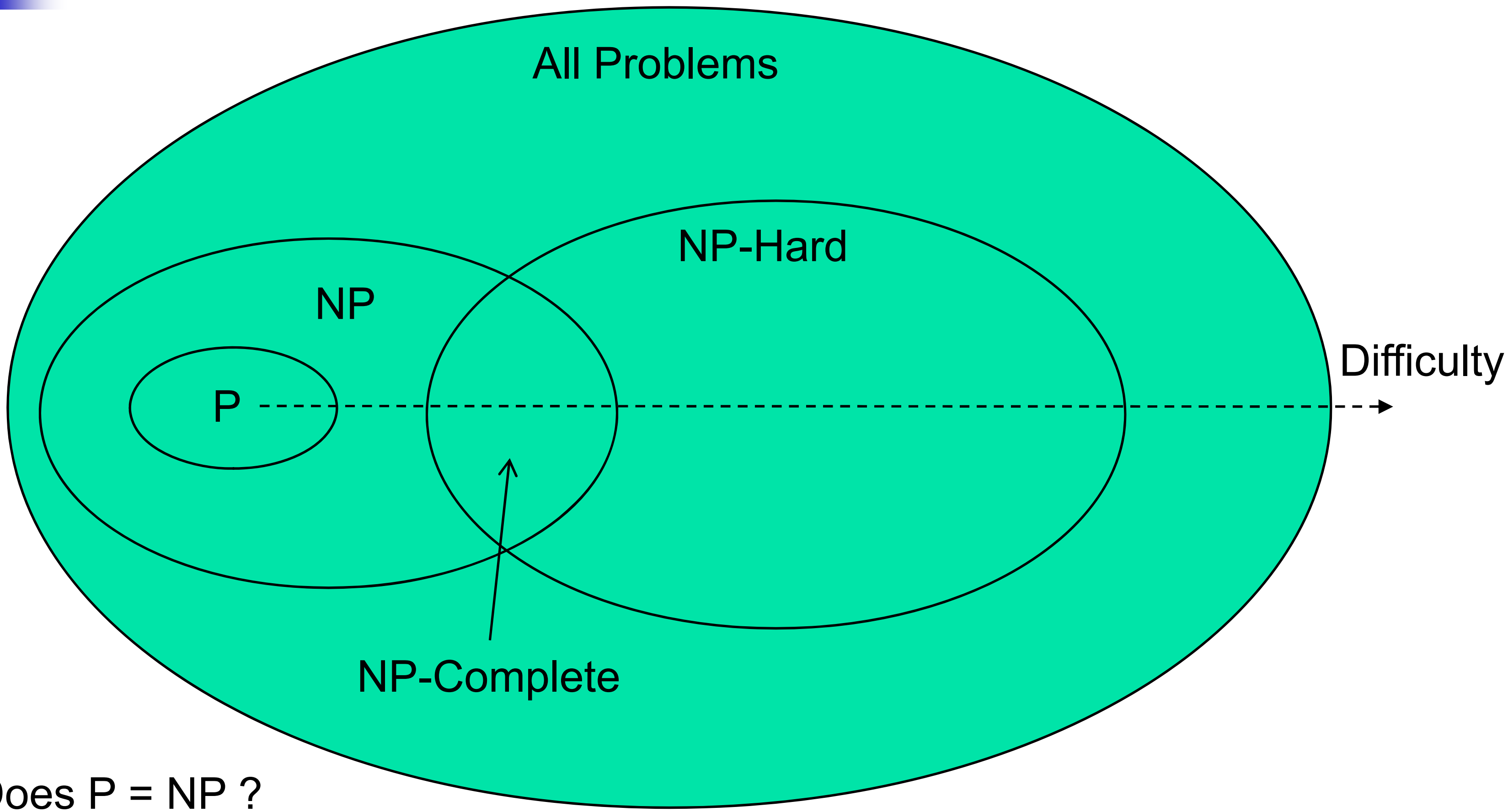
Hard Problems

- These “hard” problems belong to a class of problems that all have the same level of complexity
- Called “NP-Complete” problems
- Whether any of these problems can be solved in polynomial time is one of the biggest open questions in computer science
- Either all of these problems have polynomial-time solutions, or none of them do

Hard-est Problems

- Some problems are impossible to solve with today's computers
- Called "undecidable" problems
- E.g., the halting problem
  - Can we write a program that will determine if another program will terminate?
  - Answer: No

Classes of Hard Problems



Does P = NP ?

NP Problems

- Non-deterministic Polynomial-time (NP) problems are decidable, but can take a long time to solve
- Non-deterministic problems offer choices to the computer at each step of the solution
  - E.g., foreach edge in my graph, should I include it in the TSP tour or not?
  - |E| binary choices, $2^{|E|}$ solutions

NP Problems

- Another way to look at it
- Decision problem
  - Turn the problem into a yes/no problem
  - E.g., Is there a TSP tour with cost $\leq$ K?
- If a solution to a decision problem can be verified in polynomial time, then that problem is in the class NP
  - E.g., Check if tour visits each vertex once and sum of edge costs $\leq$ K

NP-Complete Problems

- Hardest problems in NP
- All problems in NP can be “reduced to” an NP-Complete problem
- “Reduced to” means
  - NP problem can be converted into a NPC problem in polynomial-time
  - Solution to NPC problem can be converted back into a solution to the NP problem

NP-Complete Problems

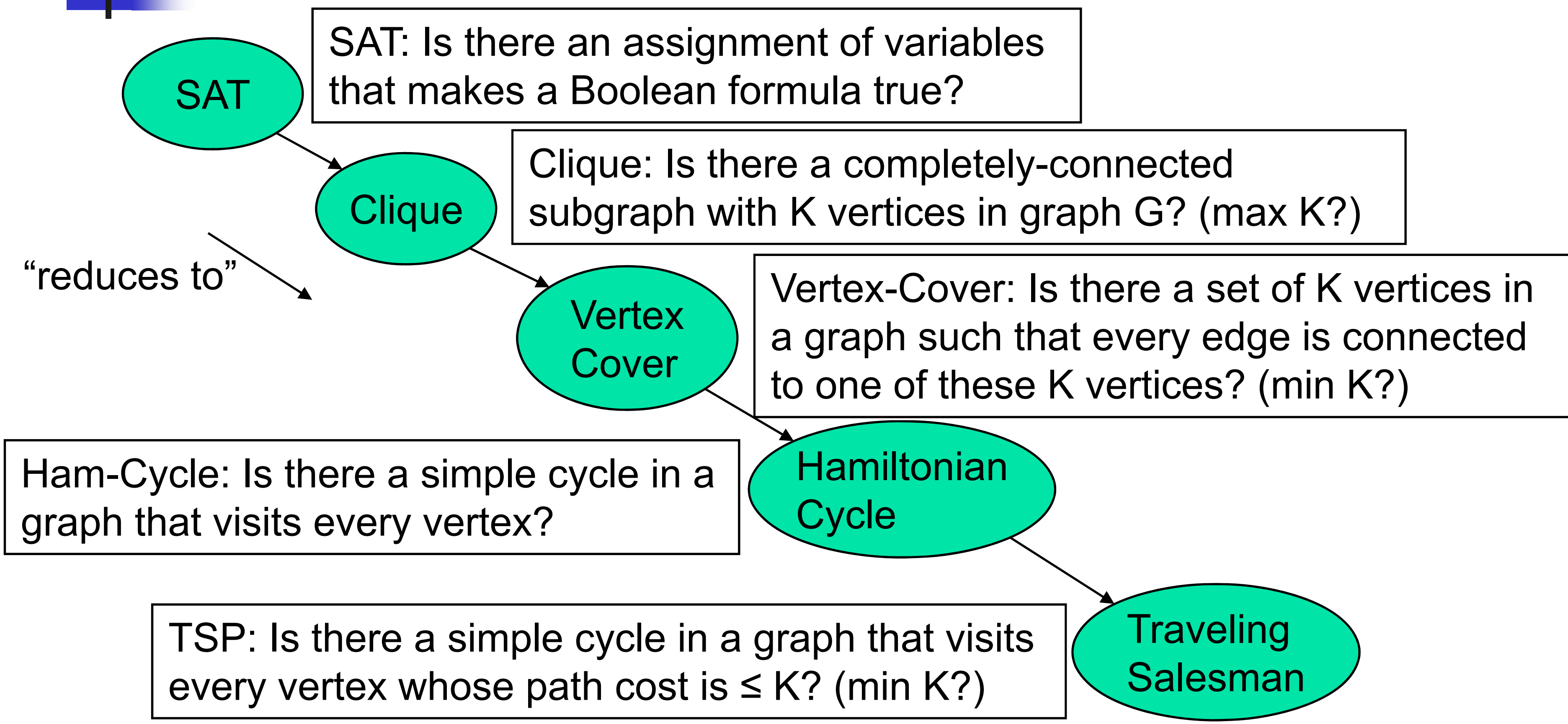
- Example: Traveling Salesman Problem
  - Given complete, weighted graph $G=(V,E)$ and integer K, is there a simple cycle that visits all vertices with total cost $\leq K$?
- Is TSP in NP? Yes
  - Given a path, we can verify (in polynomial time) if it visits all vertices and has cost $\leq K$
- Is TSP NP-Complete?
  - Need to reduce a known NP-Complete problem to TSP

NP-Complete Problems

- If we check NP-Completeness by reducing from a known NP-Complete problem, then how do we start?
- In 1971 Stephen Cook proved that the Boolean satisfiability (SAT) problem was NP-Complete using an argument from how computers work (not via reduction)
- SAT: Does a Boolean formula have a setting for variables that makes it true?
  - $x \vee (\neg y \wedge z)$?
  - $\neg x \wedge y \wedge x$?

Historical note: Cook was denied tenure at UC Berkeley in 1970, then moved to U. Toronto (still there).

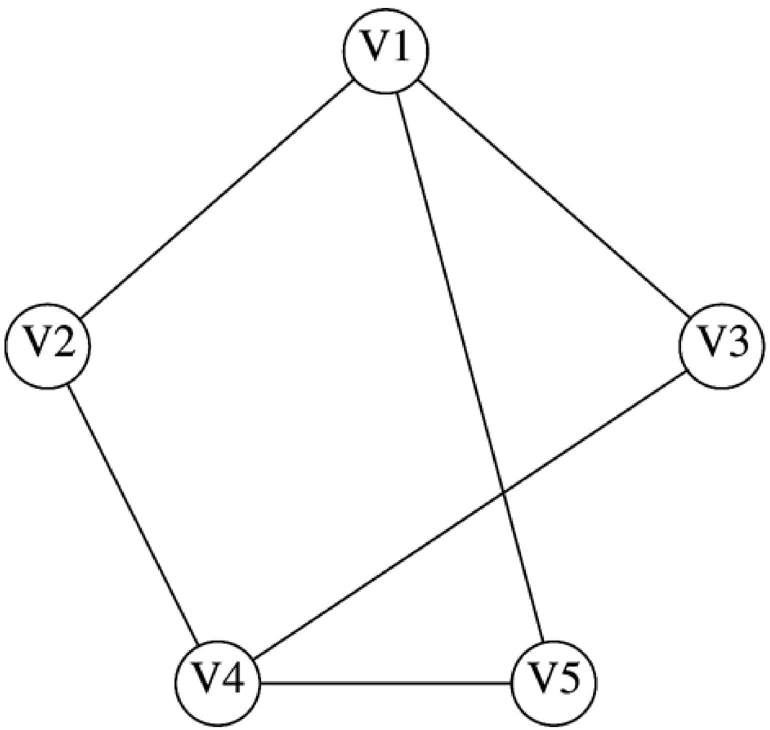
NP-Complete Problems



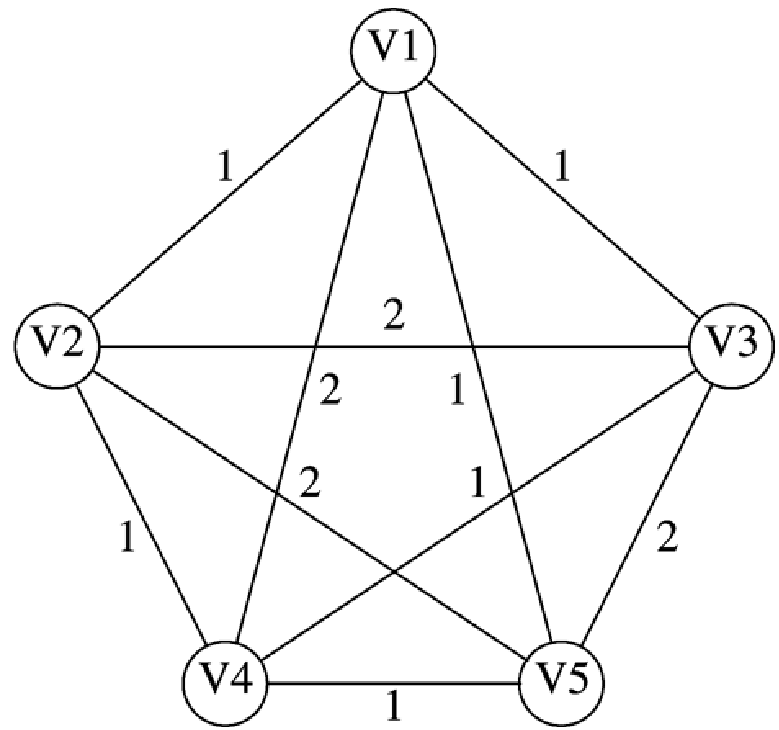
Is TSP NP-Complete?

- Assuming Hamiltonian Cycle is NP-Complete, reduce Ham-Cycle to TSP
  - Given Ham-Cycle input graph G=(V,E)
  - Create complete, weighted graph G′=(V,E′)
    - Each edge in E is in E′ with weight of 1
    - Each edge not in E is in E′ with weight of 2
  - Solve TSP with K = |V|
  - If such a TSP tour exists in G′, the same tour is a Hamiltonian Cycle in G

Reduction of Ham-Cycle to TSP



G

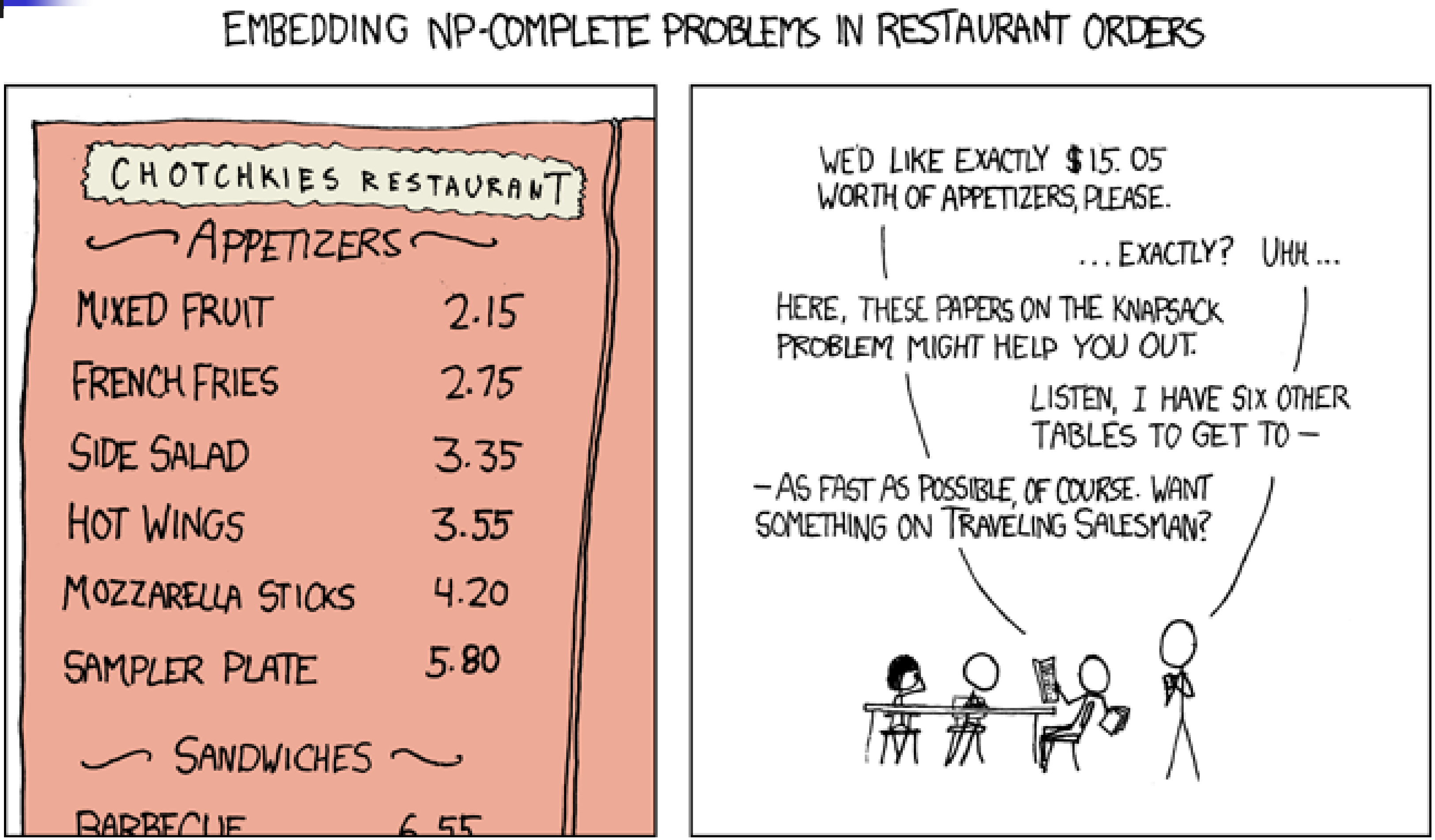


G’

NP-Complete Problems

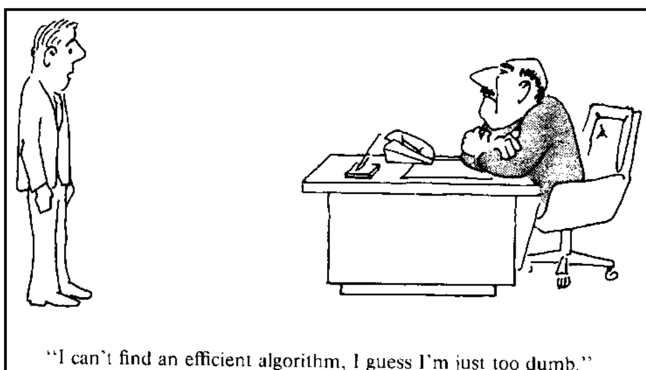
- Thus, TSP is NP-Complete
  - TSP is in NP (polynomial-time verifiable)
  - All problems in NP can be reduced to TSP
    - I.e., All problems in NP can be reduced to a problem in NP (Ham-Cycle) that can be reduced to TSP
- List of NP-Complete problems
  - http://en.wikipedia.org/wiki/List_of_NP-complete_problems

Fun with NP-Completeness

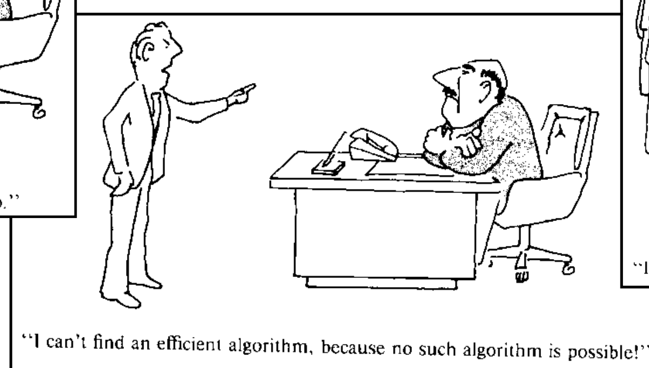


Summary

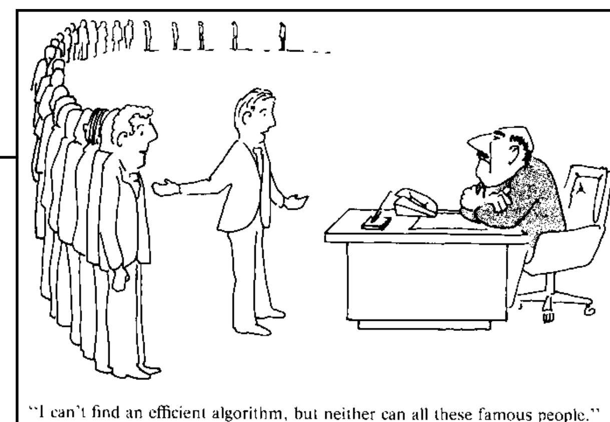
- Classes of problems helps us understand their relative difficulty
  - P, NP, NP-Complete, NP-Hard, …
- Most think P ≠ NP, but still unproven



"I can't find an efficient algorithm, I guess I'm just too dumb."



"I can't find an efficient algorithm, because no such algorithm is possible!"



"I can't find an efficient algorithm, but neither can all these famous people."

Garey & Johnson, *Computers and Intractability*, 1971.