CSE 105
THEORY OF COMPUTATION

Spring 2018

http://cseweb.ucsd.edu/classes/sp18/cse105-ab/

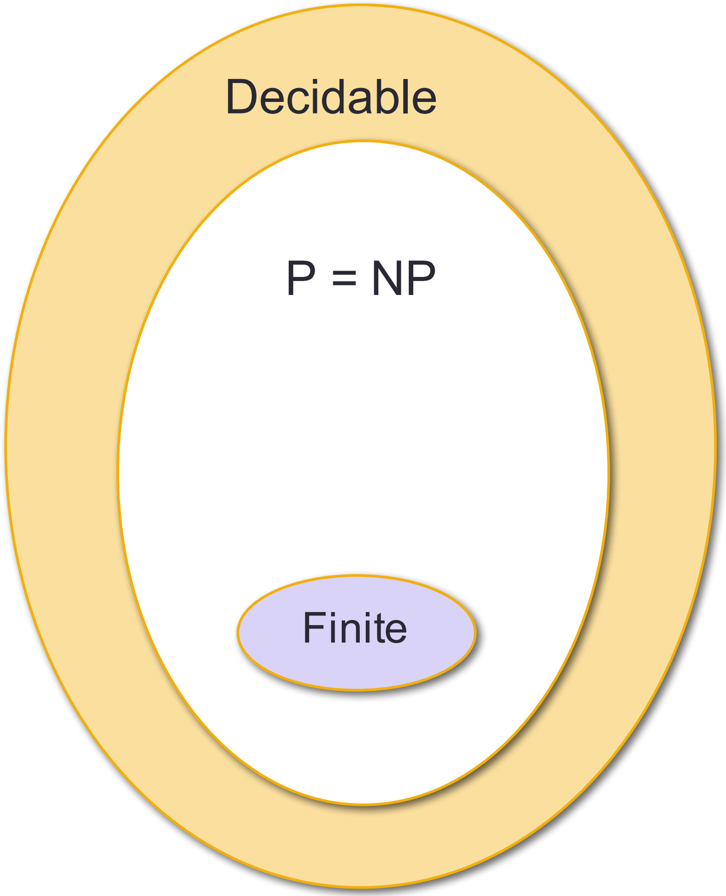
Today's learning goals

Sipser Ch 7

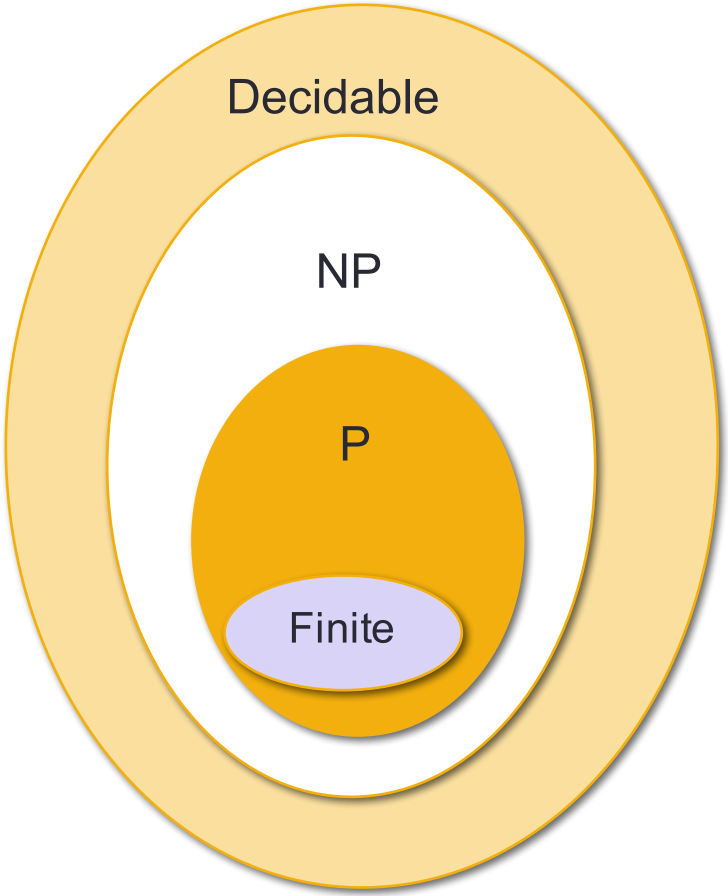
- Define NP-completeness
- Give examples of NP-complete problems
- Use polynomial-time reduction to prove NP-completeness

- Section 7.4, 7.5: NP-completeness

Start review!



or



P vs. NP

Problems in P	Problems in NP
(Membership in any) regular language	**Any problem in P**
(Membership in any) CFL	HAMPATH
PATH	CLIQUE
A_{DFA}	VERTEX-COVER
E_{DFA}	TSP
EQ_{DFA}	SAT
Addition, multiplication of integers	…

How to answer P = NP ?

Are there **hardest** NP problems?

If so, finding an efficient solution for one of them would guarantee that all NP problems have efficient solutions.

Reductions to the rescue

Sipser p. 299-305

1970s Stephen Cook and Leonid Levin indepdendently and in parallel lay foundations of **NP-completeness**

Definition A language B is **NP-complete** if (1) it is in NP and (2) every A in NP is polynomial-time reducible to it.

Problem A polynomial-time reduces to problem B means there is a polynomial-time computation function f such that, for all x, x is in A iff f(x) is in B.

Reductions to the rescue

Sipser p. 299-305

1970s Stephen Cook and Leonid Levin indepdendently and in parallel lay foundations of **NP-completeness**

Definition A language B is **NP-complete** if (1) it is in NP and (2) every A in NP is polynomial-time reducible to it.

Consequence If an NP-complete problem has a polynomial time solution then **all** NP problems are polynomial time solvable.

Reductions to

1970s Stephen Cook and
parallel lay foundations o

Definition A language B
(2) every A in NP is polyn

What would prove that P = NP?

A. Showing that a problem solvable by brute-force methods has a nondeterministic solution.
B. Showing that there are two distinct NP-complete problems.
C. Finding a polynomial time solution for an NP-complete problem.
D. Proving that an NP-complete problem is not solvable in polynomial time.
E. I don't know

Consequence If an NP-complete problem has a polynomial time solution then **all** NP problems are polynomial time solvable

3-SAT

Rosen p. 299

Cook-Levin Theorem: 3-SAT is NP-complete.

$$(x \vee \bar{y} \vee \bar{z}) \wedge (\bar{x} \vee y \vee z) \wedge (x \vee y \vee z)$$

Are other problems NP complete?

To prove that X is NP-complete

From scratch: Prove it is NP, and that all NP problems are polynomial-time reducible to it.

Using reduction*:* Show that a (known-to-be) NP complete problem reduces to it.

3SAT polynomial-time reduces to CLIQUE

Sipser p. 302

Given: Boolean formula in CNF with exactly 3 literals/clause

- AND of ORs
- args in OR clauses: var or negated var

Desired Answer: Yes if satisfiable; No if unsatisfiable

Instead transform formula to graph so that *graph has clique iff original formula is satisfiable*

3SAT polynomial-time reduces to CLIQUE

Transform 3-CNF formula with k clauses to graph G

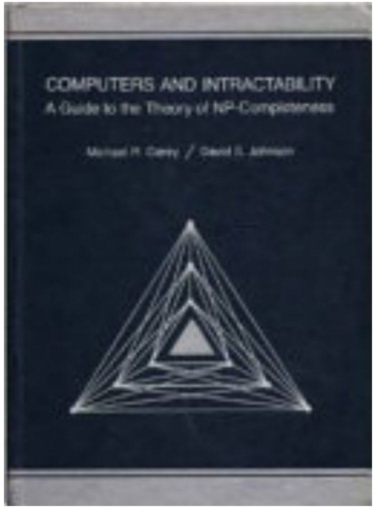
- vertices are the literals in each clause
- edges between all vertices except
  - two literals in the same clause
  - literals that are negations of one another

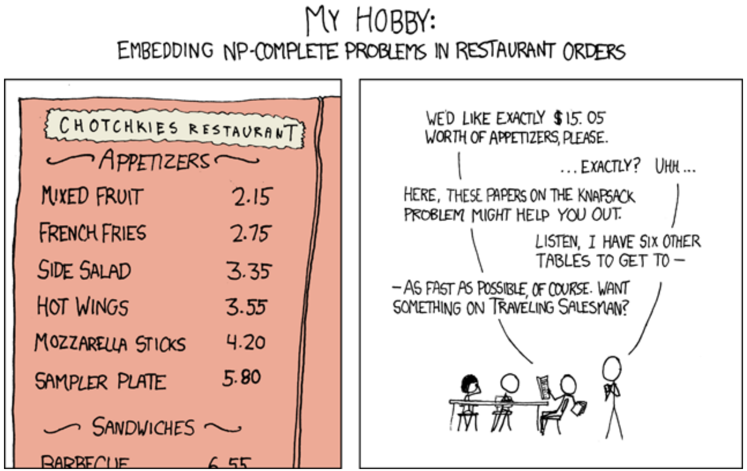
Claim: formula is satisfiable iff G has k-clique

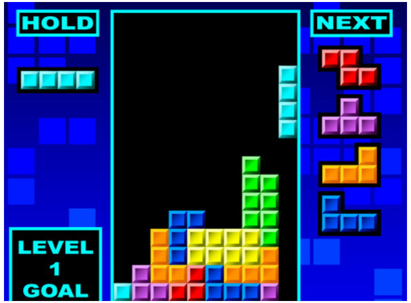
3-SAT to Clique example

$$(x \vee \bar{y} \vee \bar{z}) \wedge (\bar{x} \vee y \vee z) \wedge (x \vee y \vee z)$$

Are other problems NP-complete?







Next time

Review for final exam

Please fill out CAPE, TA evaluations.

Seat charts for final exam on Piazza.