CSE 105
THEORY OF COMPUTATION

"Winter" 2018

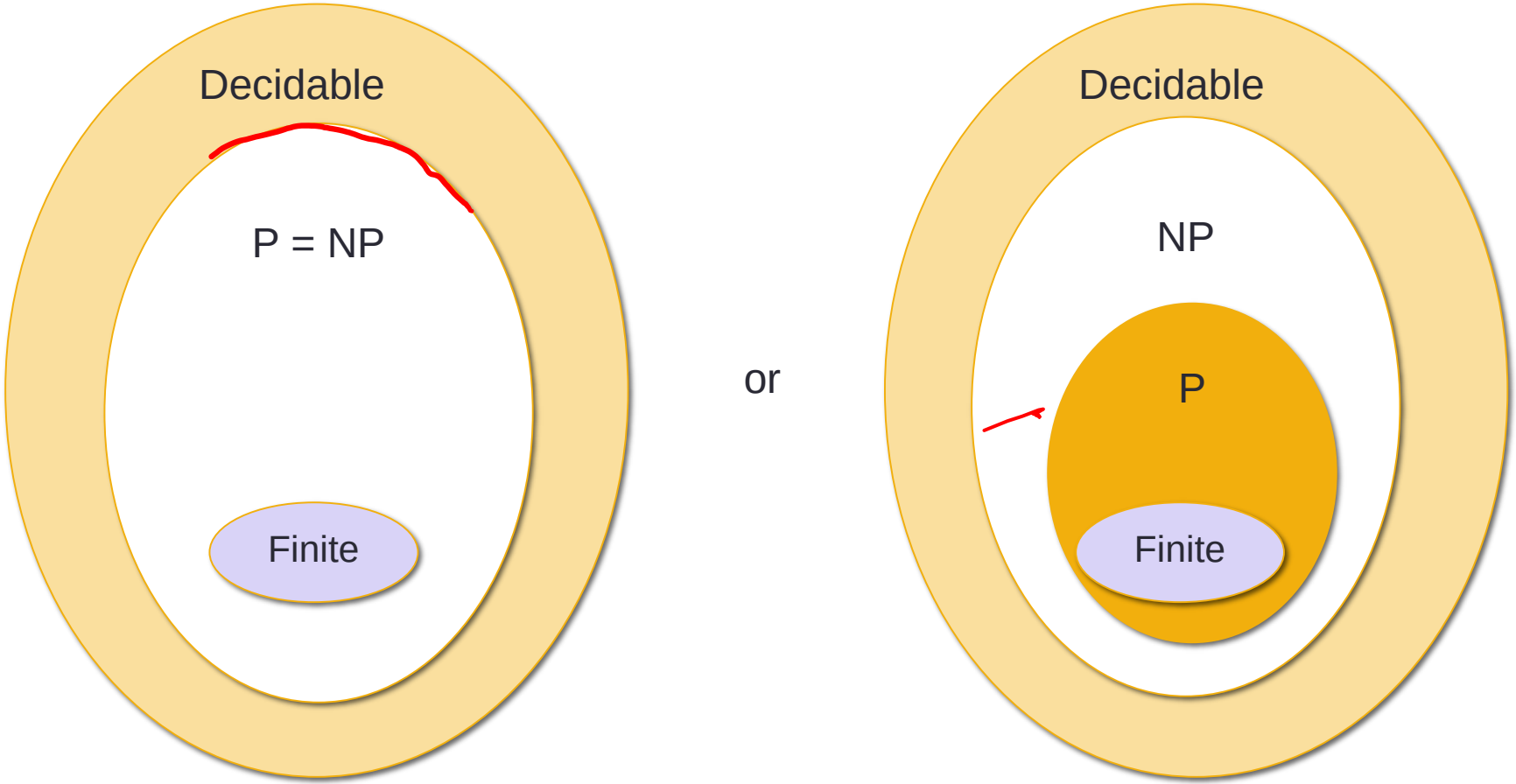
http://cseweb.ucsd.edu/classes/wi18/cse105-ab/

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


Decidable
P = NP
Finite
or
Decidable
NP
P
Finite

P vs. NP

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

a path through each vertex exactly once.

A_{DFA}

regular languages

context-free languages.

NP-complete.

How to answer P = NP ?

Suppose GENP is a problem such that

X reduces to G in polynomial time. for all $X \in NP$

this property is called

NP-completeness.

Reductions to the rescue

Sipser p. 299-305

1970s Stephen Cook and Leonid Levin independently 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.

SAT is NP-complete.

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

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.

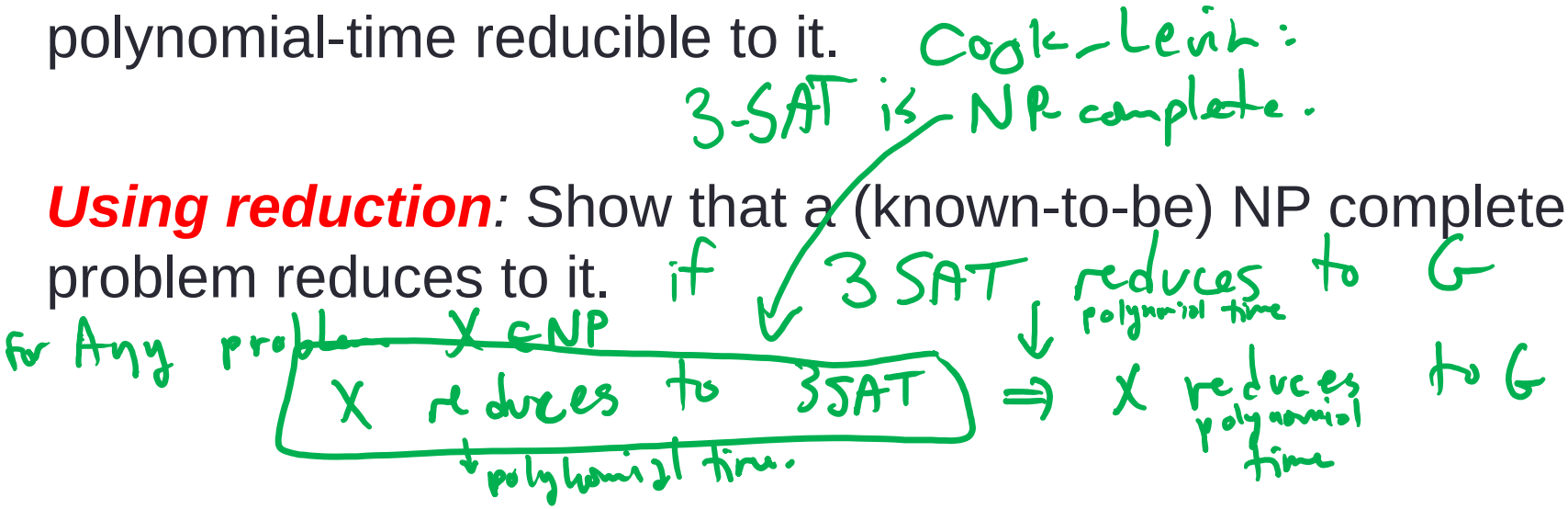
Cook-Levin Theorem: 3-SAT is NP-complete

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

3SAT reduces to Clique example

- $(x \vee \overline{y} \vee \overline{z}) \wedge (\overline{x} \vee y \vee z) \wedge (x \vee y \vee \overline{z})$

3 variables. 2^3 possible assignments.

Satisfies.

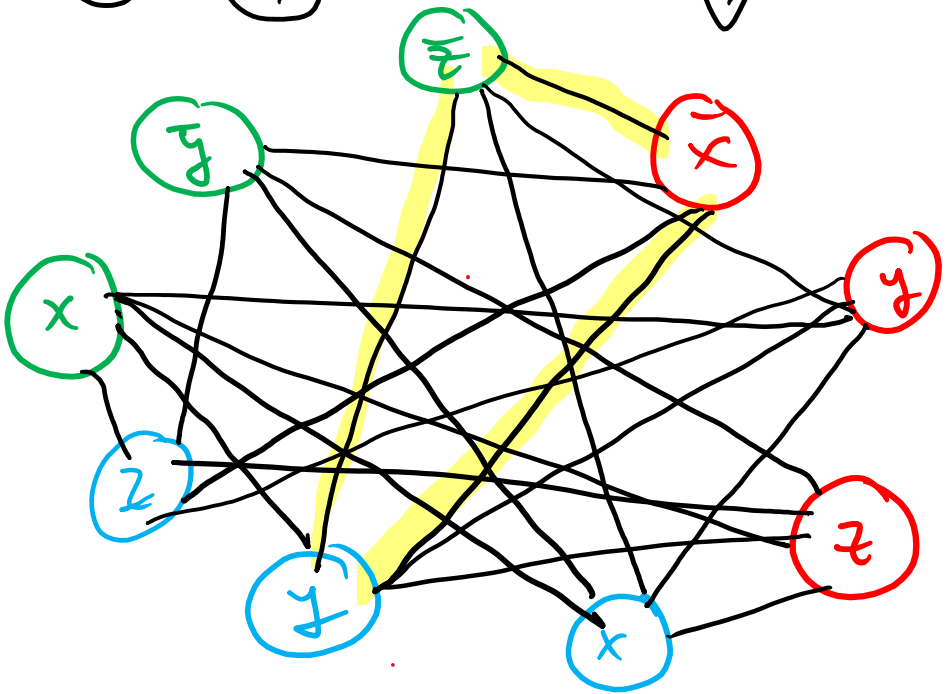
x = True
y = True
z = True

x = False
y = False
z = False

3SAT reduces to Clique example

- $(x \vee \overline{y} \vee \overline{z}) \wedge (\overline{x} \vee y \vee z) \wedge (x \vee y \vee z)$

clauses.



$x, \bar{z}, y$

x = false

z = false

y = true

⇒ CLIQUE ∈ NP-complete.

Brief history

The famous logician Kurt Godel asked the famous computer scientist, mathematician, and economist John von Neumann the P vs. NP question in a private letter, written shortly before von Neumann's death.

in the soviet union

- S.V. Yablonksi invents the term
- ``perebor'' or ``brute force search''

to describe the combinatorial explosion limiting algorithms, especially for circuit design problems (1959)

Matchings and P vs np

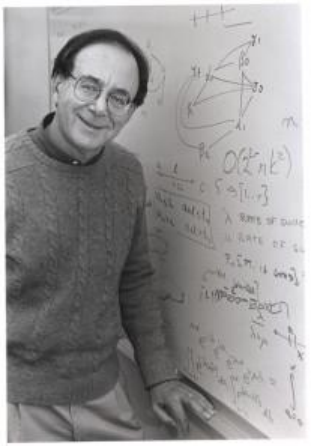
- In 1965, Jack Edmonds gives the first polynomial time algorithm for perfect matching on general graphs. To explain the significance to referees, he introduces a section defining P, NP and posing the P vs. NP question.

NP-completeness

- In 1971 , Steve Cook defines NP-completeness and proves that several problems from logic and combinatorics are NP-complete, Meaning that P=NP If and only if any of These problems are polynomial time solvable.

Plethora of Np-complete problems

- Following Cook's work,
- Richard Karp showed that a large number of the most Important optimization problems from all sub-areas (scheduling, graph theory, Number theory, logic, Puzzles and games, packing, Coding, …) are NP-complete

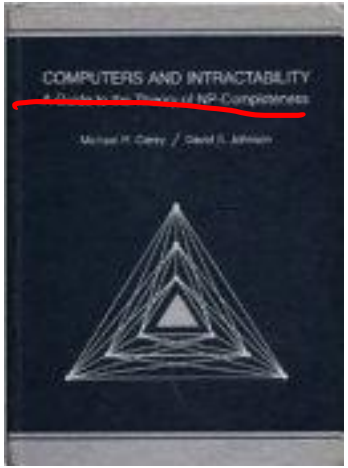


Back in the ussr

- Leonid Levin, a student of Kolmogorov's, publishes similar results to
- Cook and Karp's in his thesis,
- but needs to be careful
- to disguise what he's claiming, since it might be interpreted as Questioning earlier work on perebor.

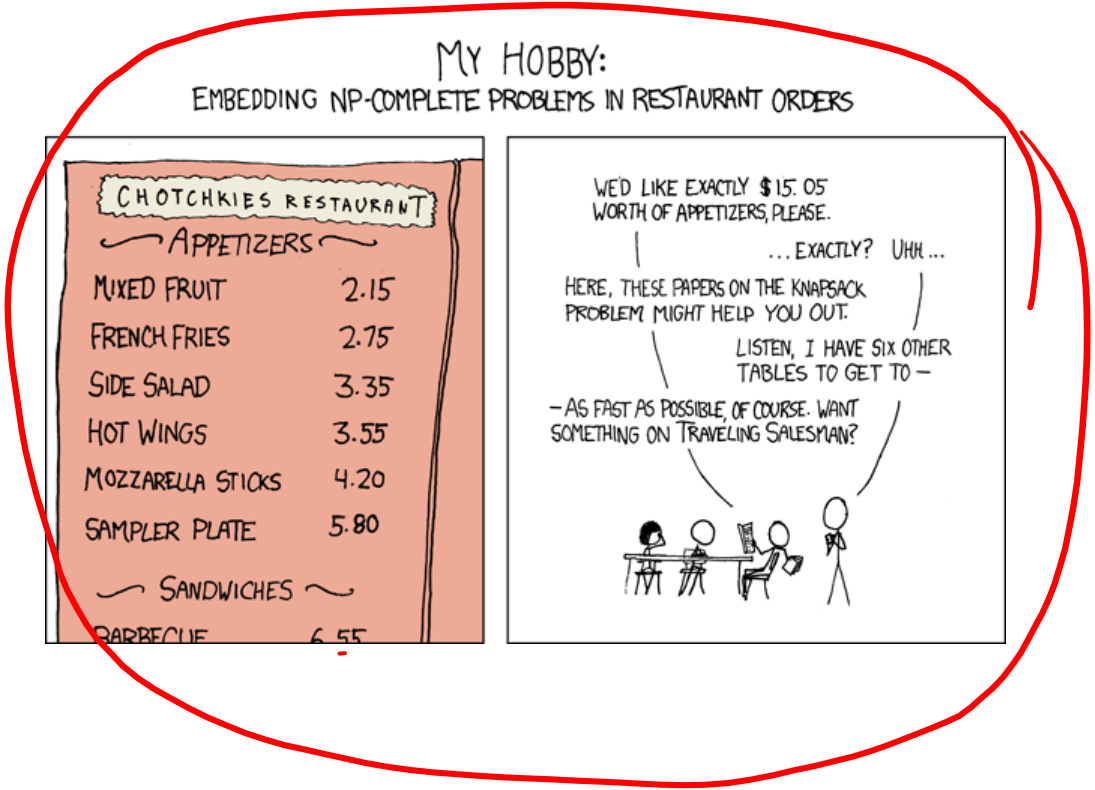
Garey and johnson

Garey and Johnson's classic Textbook (1979) includes an Appendix listing hundreds of NP-complete problems



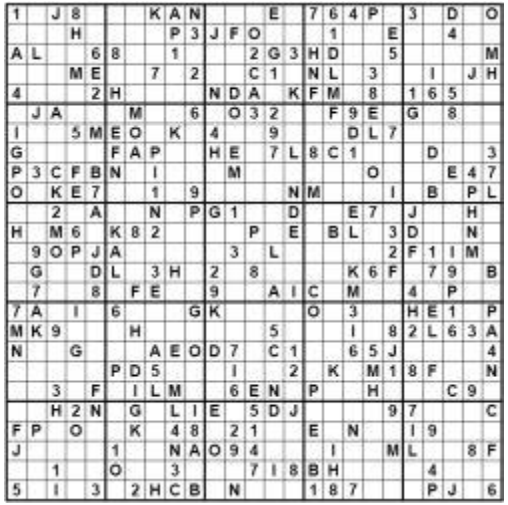
NP-complete problems everywhere

Since then, thousands
of NP-complete
Problems have been identified
in pretty much any area
With computational
Problems- physics, biology, chemistry
Economics, sociology, linguistics, games,
Engineering, …..

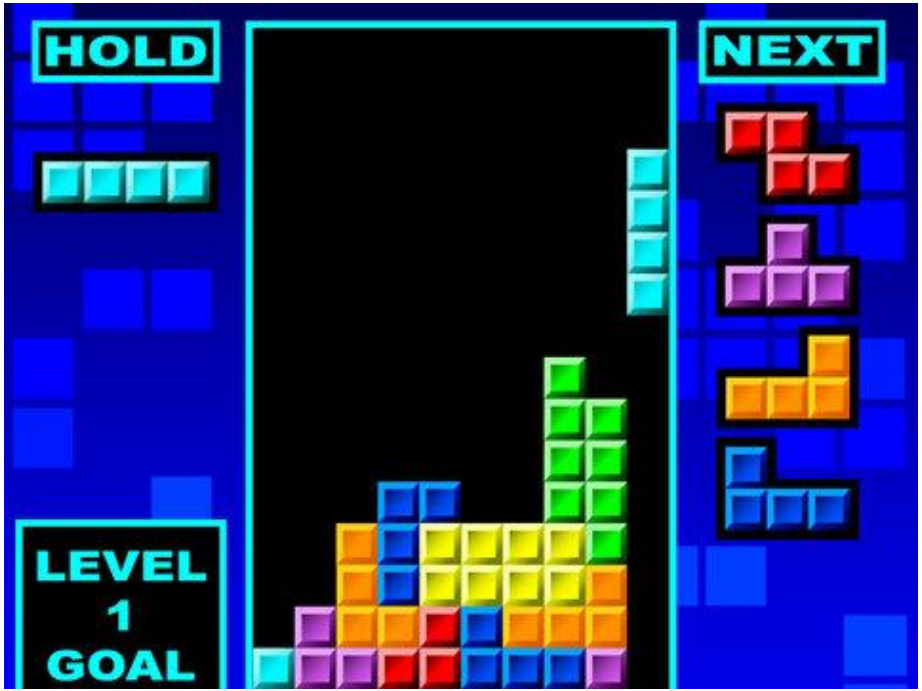


Some of our favorites

- Sudoku

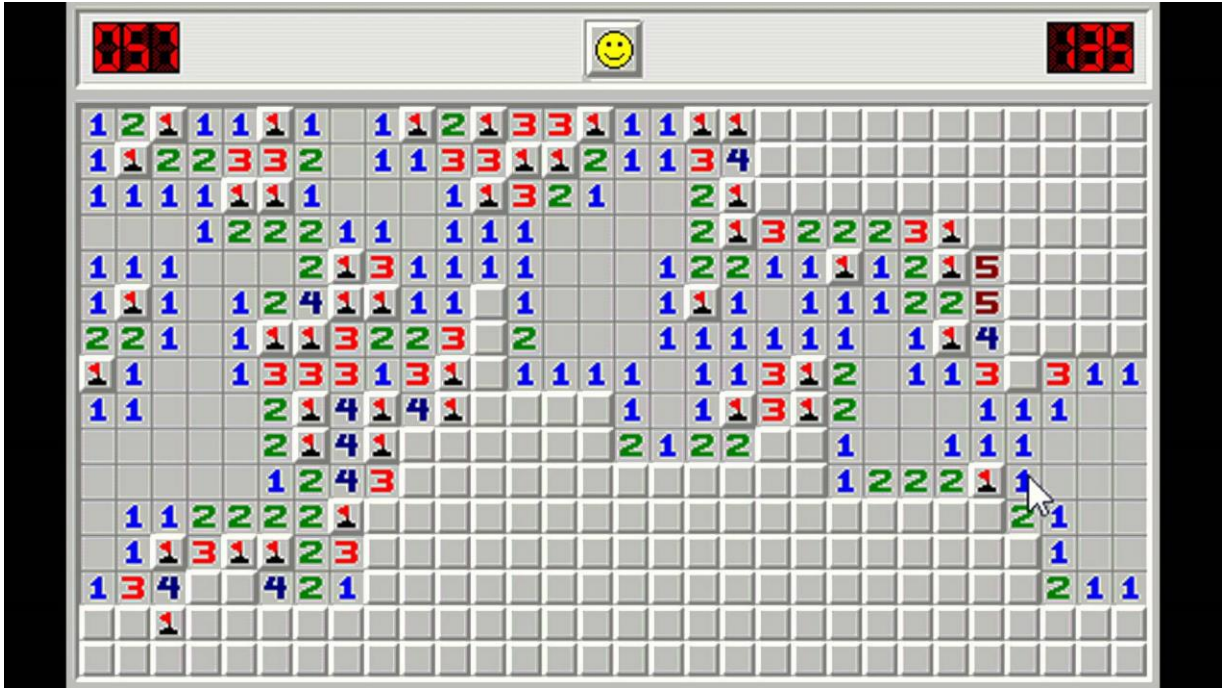


Tetris



Candy Crush

Minesweeper



Next time

Pre-class reading skim Chapter 7