Intro to Theory of Computation

Lecture 27

Last time

• Polynomial-time reductions
• NP-completeness
• Cook-Levin theorem

Today

• Examples of NP-complete problems

Sofya Raskhodnikova

4/19/2016

S. Raskhodnikova; based on slides by N. Hopper and K. Wayne.
The class NP

NP is the class of languages that have polynomial-time verifiers.
Recall: The running time of a verifier $V(\langle w, c \rangle)$ is measured only in terms of length of $w$.

If we allowed a verifier to run in time polynomial in the length of $\langle w, c \rangle$, the class NP would

A. be smaller
B. be the same
C. contain more (decidable only) languages
D. contain more (even undecidable) languages
E. none of the above
3SAT is an example of a problem that cannot be solved by an algorithm.

A. True
B. False
C. It is an open question
D. None of the above
3SAT is an example of a problem that cannot be solved by a polynomial time algorithm.

A. True
B. False
C. It is an open question
D. None of the above
A language $B$ is **NP-complete** if

1. $B \in \text{NP}$
2. $B$ is **NP-hard**, i.e.,
   every language in NP is poly-time reducible to $B$. 


![Diagram showing the relationship between P, NP, and B.]
Establishing NP-completeness

- Once we establish first "natural" NP-complete problems, others fall like dominoes.

- Recipe to establish NP-completeness of problem Y.
  - Step 1. Show that Y is in NP.
  - Step 2. Choose an NP-complete problem X (e.g., 3SAT).
  - Step 3. Prove that $X \leq_p Y$. 
SUBSET-SUM

- \( \text{SSUM} = \{ \langle S, t \rangle \mid S = \{x_1, \ldots, x_r\}, \)
- and for some \( \{y_1, \ldots, y_{r'}\} \subseteq \{x_1, \ldots, x_r\} \),
- we have \( y_1 + \cdots + y_{r'} = t \} \)

- Repetitions allowed.

- Examples: \( \langle \{5, 7, 23\}, 46\rangle \in \text{SSUM} \)
  \( \langle \{5, 15, 25\}, 46\rangle \notin \text{SSUM} \)

- \( \text{SSUM} \in \text{NP} \)
  Certificate: \( y_1, \ldots, y_{r'} \)
3SAT \leq_p \text{SUBSET-SUM}

- \text{SSUM} = \{ \langle S, t \rangle \mid S = \{x_1, \ldots, x_r\},
   \text{and for some } \{y_1, \ldots, y_{r'}\} \subseteq \{x_1, \ldots, x_r\},
   \text{we have } y_1 + \cdots + y_{r'} = t \}\}

- ``On input \langle \phi \rangle, where \phi \text{ is a 3CFN formula with variables } x_1, \ldots, x_\ell \text{ and clauses } c_1, \ldots, c_k,``

- Output a set \(S\) of numbers and target number \(t\)

- (constructed on the board) ’’
All problems below are NP-complete and hence poly-time reduce to one another!
Some NP-Complete Problems

- Six basic genres of NP-complete problems and paradigmatic examples.
  - Packing problems: SET-PACKING, INDEPENDENT SET.
  - Covering problems: SET-COVER, VERTEX-COVER.
  - Constraint satisfaction problems: SAT, 3-SAT.
  - Sequencing problems: HAMPATH, TSP.
  - Partitioning problems: 3D-MATCHING, 3-COLOR.
  - Numerical problems: SUBSET-SUM, KNAPSACK.

- Most NP problems are either known to be in P or NP-complete.
- Notable exceptions. Factoring, graph isomorphism, Nash equilibrium.
An encyclopedia of NP-complete problems