CSE 421 Algorithms

Richard Anderson

Lecture 28

NP Completeness

Announcements

- Final Exam
  - Monday, December 12, 2:30-4:20 pm, EE1 003
    - Closed book, closed notes
  - Practice final and answer key available
- HW 10, due Friday, 1:30 pm
- This weeks topic
  - NP-completeness
  - Reading: 8.1-8.8: Skim the chapter, and pay more attention to particular points emphasized in class

Algorithms vs. Lower bounds

- Algorithmic Theory
  - What we can compute
    - I can solve problem X with resources R
  - Proofs are almost always to give an algorithm that meets the resource bounds

Lower bounds

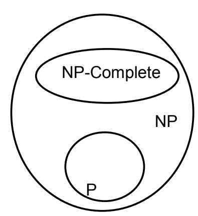
Establish evidence that a whole bunch of smart people can't do something!

- How do you show that something can't be done?

Theory of NP Completeness

Most significant mathematical theory associated with computing

The Universe



Polynomial Time

- P: Class of problems that can be solved in polynomial time
  - Corresponds with problems that can be solved efficiently in practice
  - Right class to work with “theoretically”

Polynomial time reductions

- Y Polynomial Time Reducible to X
  - Solve problem Y with a polynomial number of computation steps and a polynomial number of calls to a black box that solves X
  - Notations: $Y <_P X$

Lemma

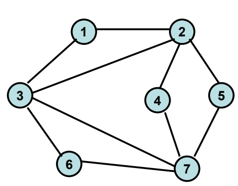
- Suppose $Y <_P X$. If X can be solved in polynomial time, then Y can be solved in polynomial time.

Lemma

- Suppose $Y <_P X$. If Y cannot be solved in polynomial time, then X cannot be solved in polynomial time.

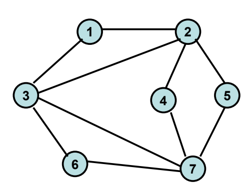
Sample Problems

- Independent Set
  - Graph G = (V, E), a subset S of the vertices is independent if there are no edges between vertices in S



Vertex Cover

- Vertex Cover
  - Graph G = (V, E), a subset S of the vertices is a vertex cover if every edge in E has at least one endpoint in S



Decision Problems

- Theory developed in terms of yes/no problems
  - Independent set
    - Given a graph G and an integer K, does G have an independent set of size at least K
  - Vertex cover
    - Given a graph G and an integer K, does the graph have a vertex cover of size at most K.

IS $<_P$ VC

- Lemma: A set S is independent iff V-S is a vertex cover
- To reduce IS to VC, we show that we can determine if a graph has an independent set of size K by testing for a Vertex cover of size n - K

Satisfiability

- Given a boolean formula, does there exist a truth assignment to the variables to make the expression true

Definitions

- Boolean variable: $x_1, \ldots, x_n$
- Term: x_i or its negation $!x_i$
- Clause: disjunction of terms
  - t_1 or t_2 or … t_j
- Problem:
  - Given a collection of clauses C1, . . ., Ck, does there exist a truth assignment that makes all the clauses true
  - (x_1 or $!x_2$), ($!x_1$ or $!x_3$), (x_2 or $!x_3$)

3-SAT

- Each clause has exactly 3 terms
- Variables $x_1, \ldots, x_n$
- Clauses $C_1, \ldots, C_k$
  - $C_j = (t_{j1}$ or t_{j2} or $t_{j3})$
- Fact: Every instance of SAT can be converted in polynomial time to an equivalent instance of 3-SAT

Theorem: 3-SAT $<_P$ IS

- Build a graph that represents the 3-SAT instance
- Vertices y_i, z_i with edges (y_i, z_i)
  - Truth setting
- Vertices u_{j1}, u_{j2}, and u_{j3} with edges (u_{j1}, u_{j2}), (u_{j2}, u_{j3}), (u_{j3}, u_{j1})
  - Truth testing
- Connections between truth setting and truth testing:
  - If $t_{jl} = x_i$, then put in an edge (u_{jl}, z_i)
  - If $t_{jl} = !x_i$, then put in an edge (u_{jl}, y_i)

Example

$C_1 = x_1$ or x_2 or $!x_3$

$C_2 = x_1$ or $!x_2$ or x_3

$C_3 = !x_1$ or x_2 or x_3

Thm: 3-SAT instance is satisfiable iff there is an IS of size n + k