Week 9, Lecture 15 on 16 October 2021 - CS1.301.M21 Algorithm Analysis and Design

NP-Completeness Theory

The Class NP

A verifier for a language A is an algorithm V, where

$$A = \{w | V \text{ accepts } \langle w, c \rangle \text{ for some string c } \}$$

NP is the class of languages that have polynomial time verifiers.

Example 1:

$$\text{CLIQUE} = \{\langle G, k \rangle | G \text{ is an undirected graph with a k-clique}\}$$

V = On input $\langle \langle G, k \rangle, c \rangle$:

1. Test whether c is a subgraph with k nodes in G.
2. Test whether G contains all edges connecting nodes in c.
3. If both pass, accept; otherwise, reject.

This verifier V is a verifier for CLIQUE.

Example 2:

$$SUBSET-SUM = \langle S, t \rangle | S = \{x_1, \ldots, x_k\}, \text{ and for some } \{y_1, \ldots, y_l\} \subseteq \{x_1, \ldots, x_k\}, \text{ we have } \Sigma y_i = t.$$

Polynomial Time Reducibility

A polynomial time reduction of A to B is a function f:

$$w \in A \iff f(w) \in B$$

i.e. if an answer w belongs to A then f is a polynomial time reduction if and only if $f(w)$ also belongs to B.

> *3SAT is a boolean satisfiability problem*

3SAT is polynomial time reducible to CLIQUE.

$$3SAT = \{\langle\phi\rangle | \phi \text{ is a satisfiable 3cnf-formula}\}$$

> *When a problem polynomial time reducible to another it really means that if a problem is polynomial time solvable then with only polynomial time effort we can also solve the other problem.*

Proof of reducibility of 3SAT to CLIQUE: Sipser, pg.302 Theorem 7.32

NP Completeness

A language B is *NP-complete* if it satisfies two conditions:

1. B is in NP, and
2. Every A in NP is polynomial time reducible to B.

Since polynomial time reducibility is a transitive property, if a language C is polynomial time reducible to B then C is also NP-Complete.

SAT (boolean satisfiability) was proved to be NP-complete. Now any problem reducible to/from then that problem is also NP-complete.