Exercise 11.1.1 b)

E 9.1

Consider the problem FALSE-SAT:

Given a boolean expression E that is false when all its variables are made false, is there some other truth assignment that makes E false, besides all-false?

Decide whether the problem is in NP or coNP.

Describe its complement.

If the problem or its complement is NP-complete, prove it.

Proof:

The problem is NP-complete.

- In NP: given a boolean expression E, we need to check:
  1) that E is false when all variables are assigned false
  2) that there is some other truth-assignment making E false

  (1) can be done in poly-time by a DTM
  (2) can be done in poly-time by a NTM
  guess a truth-assignment T different from all false, and answer yes if under T, E evaluates to false

- NP-hard: by a reduction from SAT
  Let E be a boolean expression with variables $x_1, \dots, x_n$.
  We construct an expression E' s.t. E ∈ SAT iff E' ∈ FALSE-SAT
  1) Test if E is true when all variables are false (polynomial)
  If so, E ∈ SAT, and we convert it to a fixed expression that is in FALSE-SAT, e.g. $x \wedge y$.

E9.2

2) Otherwise, let E' be $\neg E \wedge (x_1 \vee x_2 \vee \cdots \vee x_n)$.

Clearly, the reduction is poly-time.

We have that E' is false when all of $x_1, \ldots, x_n$ are false.

Notice that in case (2), E is false when all variables are false.

Hence, if $E \in SAT$, then it is satisfied by a truth assignment T different from all-false.

Thus, $\neg E$ is made false by T, and $E' \in$ FALSE-SAT.

Conversely, if $E' \in$ FALSE-SAT, then since $x_1 \vee \cdots \vee x_n$ is false only for the all-false truth assignment, there must be some other truth-assignment T that makes $\neg E$ false. Then T makes E true, and $E \in SAT$.

Exercises on problems in P, NP, and NP-complete

Exercise

Consider the following optimization version of SAT:

MAXSAT: Input: a propositional formula F in CNF, and an integer k

Output: yes, if there is a truth assignment that satisfies at least k clauses of F

no, otherwise

What is the complexity of MAXSAT?

a) MAXSAT $\in$ NP: immediate, by the following NP algorithm

1) guess a truth assignment α (non-deterministic polynomial)
2) count the # of clauses satisfied by α, and answer yes iff it is $\geq k$ (deterministic polynomial)

b) MAXSAT is NP-hard

This follows from the fact that CSAT is a special case of MAXSAT.

Formally, we can polynomially reduce CSAT to MAXSAT, i.e.

$$\text{SAT} \leq_{poly} \text{MAXSAT}$$

Given an instance F of CSAT, we construct an instance (F, k) of MAXSAT, where k is the # of clauses of F.

Obviously, k can be obtained in polytime from F, and

$$F \in \text{CSAT} \iff (F, k) \in \text{MAXSAT}$$

□

E3.4

Exercise 4

Consider the following problems:

1) Vertex-cover (VC)

Given an undirected graph $G=(V,E)$ and an integer $k \geq 2$, is there a subset C of V with $|C| \leq k$ such that C covers all edges of G (i.e., for each edge $\{v_i, v_j\} \in E$ with $v_i \neq v_j$, $\{v_i, v_j\} \cap C \neq \emptyset$).

2) Independent-set (IS)

Given an undirected graph $G=(V,E)$ and an integer $k \geq 2$, is there a subset C of V with $|C| \geq k$ such that for all $v_i, v_j \in C$ with $v_i \neq v_j$, $\{v_i, v_j\} \notin E$

3) Clique

Given an undirected graph $G=(V,E)$ and an integer $k \geq 2$, is there a subset C of V with $|C| \geq k$ such that for all $v_i, v_j \in C$ with $v_i \neq v_j$, $\{v_i, v_j\} \in E$.

Show that VC, IS, and Clique can be reduced to each other in polynomial time.

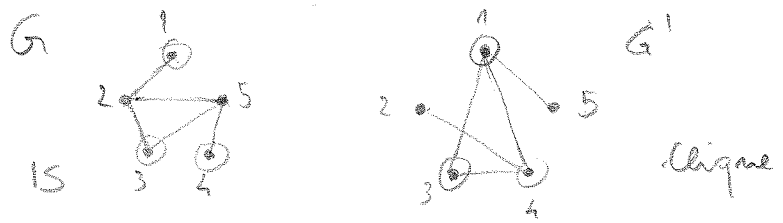
N.B. In the definitions of VC, IS, and Clique we have ignored self-loops (since we required $v_i \neq v_j$).

$IS <_{poly}$ Clique

Given an instance (G, k) of IS, we construct an instance (G', k') of Clique as follows:

$k' = k$
Let $G = (V, E)$.
Then $G' = (V, E')$, where $E' = V \times V \setminus E$
(i.e., the edges of E' are obtained by connecting all pairs of nodes that are not connected in E.)

E.g. G



The reduction works because the maximum independent set of G is precisely the maximum clique in the complement graph of G.

$VC <_{poly}$ IS

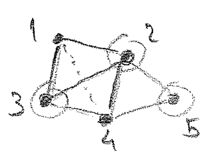
Given an instance (G, k) of VC, we construct an instance (G', k') of IS as follows:

$k' = |V| - k$

$G' = G$

The reduction works because the vertexes in a vertex-cover C cover all edges of G. Hence the set $V \setminus C$ must have no edges between its elements, and is thus an independent set

E.g.



The marked nodes {2,3,5} are a VC of size 3. Hence, there cannot be an edge {1,4}

E 9.6

Clique $\leq_{poly}$ VC

Given an instance (G, k) of Clique, we construct an instance (G', k') of VC as follows:

$k' = |V| - k$

Let $G = (V, E)$.

Then $G' = (V, E')$, where $E' = V \times V \setminus E$