CS3230
Tutorial 12

1. 2-SAT denotes the variant of satisfiability where each clause has at most two literals.

   Show that 2-SAT is in P.

   Ans: Suppose $V = \{x_1, x_2, \ldots, x_n\}$ is the set of variables, and $C = \{s_1, s_2, \ldots, s_m\}$ is the set of clauses.

   Construct a directed graph G as follows. Vertices of the graph are $\{x_i, \neg x_i : 1 \le i \le n\}$. There is an edge from a literal ℓ to ℓ', iff there is a clause of the form ($\neg\ell$ or ℓ') or (ℓ' or $\neg\ell$). Here, we take $\neg\neg x_i$ to be x_i. Intuitively, edge from ℓ to ℓ' denotes that ℓ being true implies that ℓ' is true.

   For each literal ℓ, let $closure(\ell) = \{\ell' : \text{there is a directed path in } G \text{ from } \ell \text{ to } \ell'\}$. Note that $closure(\ell)$ can be computed in polynomial time from the graph G described above.

   Intuitively, ℓ being true implies that each literal in $closure(\ell)$ is true. Note that $\ell \in closure(\ell)$.

   Note that if $closure(\ell)$ contains x_i and $\neg x_i$, for some i (that is, $closure(\ell)$ contains contradictory literals), then ℓ cannot be true.

   On the other hand, if $closure(\ell)$ does not contain any contradictory literals, then setting ℓ to be true does not hurt satisfiability of the clauses. This is so because setting each member of $closure(\ell)$ to be true divides C into two parts: C_1 and C_2, where each clause in C_1 contains a literal from $closure(\ell)$ and each clause in C_2 does not contain any literal from either $closure(\ell)$ or $oppclosure(\ell) = \{\neg\ell' : \ell' \in closure(\ell)\}$.

   Thus, the 2-SAT instance given by (V, C) is satisfiable iff there does not exist a variable x_i such that $closure(x_i)$ contains contradictory literals and $closure(\neg x_i)$ contains contradictory literals. It follows that 2-SAT is in P.

   [In fact, the last paragraph above can be simplified to say that there is no variable x_i such that $closure(x_i)$ contains $\neg x_i$ and $closure(\neg x_i)$ contains x_i.]

2. Consider the following problem:

   INPUT: (a) A finite universal set U,

   (b) a collection $\mathcal{S}$ of subsets of U, and

   (c) a number m.

   QUESTION: Do there exist m sets $S_1, S_2, \ldots, S_m$ in $\mathcal{S}$ such that for $1 \le i < j \le m$, $S_i \cap S_j = \emptyset$?

   Show that the above problem is NP-complete.

Ans: It is easy to verify that the above problem is in NP. Certificates would be m sets $S_1, S_2, \ldots, S_m$. Checking of the certificate is to check that the m sets $S_1, S_2, \ldots, S_m$ are in $\mathcal{S}$, and that for $1 \leq i < j \leq n$, $S_i \cap S_j = \emptyset$.

To show NP-hardness, we reduce 3DM to the Q2-problem.

Suppose the 3DM instance is X, Y, Z, S. Without loss of generality assume X, Y, Z are pairwise disjoint.

Then, convert it to Q2-instance by having $U = X \cup Y \cup Z$, $m = |X| = |Y| = |Z|$, and $\mathcal{S} = \{\{x, y, z\} : (x, y, z) \in S\}$.

Now, if 3DM instance has a matching S', where the elements of the matching are (x_1, y_1, z_1), ..., (x_m, y_m, z_m). Then choosing $S_i = \{x_i, y_i, z_i\}$, gives a solution to Q2-instance.

On the other hand, if the Q2-instance has a solution $S_1, S_2, \ldots, S_m$, where $S_i = \{x_i, y_i, z_i\}$, then $(x_1, y_1, z_1), \ldots, (x_m, y_m, z_m)$ gives a matching for the 3DM-instance.

3. Consider the following decision problem:

   INPUT: Two grapha $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$, and a number K.

   QUESTION: Do there exist subgraph's G_1' of G_1 and G_2' of G_2 such that G_1' and G_2' are isomorphic and G_1' (and thus G_2') has at least K edges?

   Show that the above problem is NP complete.

   Ans: It is easy to verify that the above problem is in NP. Certificate would be V_1', E_1', V_2', E_2' along with a mapping f from V_1' to V_2'.

   Checking would be to verify that (i) $V_1' \subseteq V_1$, $V_2' \subseteq V_2$, (ii) E_1', E_2' are subsets of E_1 and E_2 and have only edges involving V_1' and V_2' respectively, (iii) $|E_1'| = K$, and (iv) f witnesses an isomorphism from (V_1', E_1') to (V_2', E_2').

   To see that the above problem is NP-hard reduce Hamiltonian Circuit problem to the above problem as follows. Suppose $G = (V, E)$ is an instance of Hamiltonian circuit problem. Create an instance of this question by taking $G_1 = G$, $G_2 = (V, C)$, and $K = |V|$, where for $V = \{v_1, v_2, \ldots, v_n\}$, $C = \{(v_1, v_2), (v_2, v_3), \ldots, (v_{n-1}, v_n), (v_n, v_1)\}$.

   Now, if the graph G has an Hamiltonian Circuit, then clearly both graphs have a subgraph which is G_2. On the other hand, if G_1, G_2 have a common subgraph with $|V|$ edges, then then it means G_2 is a subgraph of G_1, and thus G_1 has an Hamiltonian Circuit.

4. Consider the following decision problem.

   > Input: A set V of variables, a set C of clauses over V such that each clause has at most 3 literals.
   > Question: Does there exist a truth assignment to the variables in V such that each clause in C has exactly one literal true?

   Prove that the above problem is NP-complete.

   Ans: To see that the above problem is in NP, consider certificates as truth assignment to the variables. For verification of the certificates, just check if the truth assignment to the variables makes each clause have exactly one true literal.

To show that the above problem is NP-hard, consider a reduction from 3-SAT as follows.

Suppose (V, C) is an instance of 3-SAT. We reduce it to (V', C') an instance of Q4 as follows. Let $V' = V \cup \{W_{i,c} \mid i \leq 5 \text{ and } c \in C\}$, where $W_{i,c}$ are new variables. For each $c = (x \vee y \vee z) \in C$, C' contains the clauses

$(x\ W_{0,c}\ W_{1,c})$, $(y\ W_{0,c}\ W_{2,c})$, $(z\ W_{3,c})$, $(W_{0,c}\ W_{3,c}\ W_{4,c})$, and $(W_{1,c}\ W_{2,c}\ W_{5,c})$.

We claim that there exists a satisfying truth assignment for (V, C) iff there exists a truth assignment to variables in V' such that each clause in C' has one and only one true literal.

Suppose TA is a satisfying truth assignment for (V, C). Truth values assigned to variables in V' which appear in V are as in TA. For $c = (x\ y\ z)$, $W_{i,c}$ are assigned truth values as follows: $W_{3,c}$ is true iff z is false; $W_{0,c}$ is true iff both x and y are false; $W_{1,c}$ is true iff x is false and y is true; $W_{2,c}$ is true iff y is false and x is true. $W_{4,c}$ is true iff both $W_{0,c}$ and $W_{3,c}$ are false. $W_{5,c}$ is true iff both $W_{1,c}$ and $W_{2,c}$ are false.

It is easy to verify that each clause in C' has exactly one true literal.

Suppose TA is a truth assignment for variables in V' such that each clause in C' has exactly one true literal. Then, we claim that TA restricted to variables in V is a satisfying truth assignment for (V, C). To see this suppose by way of contradiction that, for some clause $c = (x\ y\ z)$ in C, none of the literals are true. Then consider the clauses $(x\ W_{0,c}\ W_{1,c})$, $(y\ W_{0,c}\ W_{2,c})$, $(z\ W_{3,c})$, $(W_{0,c}\ W_{3,c}\ W_{4,c})$, and $(W_{1,c}\ W_{2,c}\ W_{5,c})$ of C'. Note that $W_{3,c}$ must be true (due to $(z\ W_{3,c})$). Thus, $W_{0,c}$ must be false (due to $(W_{0,c}\ W_{3,c}\ W_{4,c})$). It follows that both $W_{1,c}$ and $W_{2,c}$ must be true (due to $(x\ W_{0,c}\ W_{1,c})$, and $(y\ W_{0,c}\ W_{2,c})$). But then, $(W_{1,c}\ W_{2,c}\ W_{5,c})$ has two true literals.

5. Suppose X and Y are two sequences of the same length, and both X and Y have a as their third character. Then can we claim that the longest common subsequence of X and Y also has a in it?

   Ans: The claim is false.

   Counterexample:

   Sequence 1: *bbacc*

   Sequence 2: *ccabb*

   Then, the maximal common subsequence is either *bb* or *cc*. However any common subsequence which contains a is of length 1.