# PROBLEM SET 8 – Manav Bilakhia

CSC 250, Spring 2022
Assigned: Wednesday, Week 9
Due: Wednesday, Week 10

Aaron G. Cass
Department of Computer Science
Union College

## NP-COMPLETENESS

1. In the NON-TAUTOLOGY problem, we are given a boolean formula $F$ with literals, connectives from the set $\{\wedge, \vee, \neg, \rightarrow, \leftrightarrow\}$, and parentheses. The question is: Does $F$ have a truth assignment that makes it equal to 0?

   Prove that NON-TAUTOLOGY is NP-Complete.

**Answer:**

In order to prove that NON-TAUTOLOGY [NT] is NP-Complete we must show that:

- **The NT $\in$ NP:**
  We can create a Boolean formula $f$ such that a truth assignment will make it equal to 0. We then recursively replace each pair wise operation with its value i.e. 0 or 1 such that the size of the formula is reduced by almost half after every replacement. This recursion runs in polynomial time. If the final replacement yields 0 then our solution certificate is verified. If not then the the solution certificate is incorrect.

  We were able to verify the solution certificate in polynomial run time.

- **$\forall$ A $\in$ NP, A $\leq_p$ NT:**
  We know that the SAT problem is NP-hard, We want to show that SAT $\leq_p$ NT. SAT takes formula f as input. Let us compute $f'$ such that $f' = \neg f$. We should be able to perform this operation in linear time. Then pass $f'$ as the input for NT and return the output for NT. The SAT problem returns true iff there exists an input such that the formula equals 1. Where as the NT problem returns true iff there exists an assignment such that the formula produces 0. Hence providing the NT problem with a negated formula would solve the SAT problem. Therefore SAT $\leq_p$ NT and NT is NP-Hard.

  We have shown that NT is NP-Hard and its solution certificate is can be verified in polynomial time. Hence NT is NP Complete.

2. **Subgraph-Isomorphism.** [From [1], #34.5-1, pg 1017] The subgraph-isomorphism problem takes two graphs $G_1$ and $G_2$ and asks whether there exists a subgraph $G_2'$ of $G_2$ such that $G_1$ is isomorphic to $G_2'$. Prove that the subgraph-isomorphism problem is NP-Complete.

**Answer:**

In order to prove that Subgraph-Isomorphism [SI] is NP-Complete we must show that:

- **The SI $\in$ NP:**
  In order to show that SI $\in$ NP, we must be able to verify the solution certificate for SI in polynomial time. Let $G_2'$ be a sub-graph of $G_2$. Therefore we now also know the mapping between the vertices of $G_1$ and $G_2'$. In order to check if $G_1$ is isomorphic to $G_2'$, We must check if the mapping is a bijection and check that for every edge in (u,v) in $G_1$, there is an edge (f(u),f(v)) present in $G_2'$. This will take polynomial time.

- **$\forall$ A $\in$ NP, A $\leq_p$ SI**
  For this proof let us choose the Clique problem [C]. We know that Clique is a NP-Hard problem. Therefore we must show that C $\leq_p$ SI. We must be able to write the algorithm of Clique using Subgraph Isomorphism as a sub routine.

The Clique problem takes graph G and size k as input. It returns true if a clique of size k exists in G. Let $G_1$ be a graph of k vertices and $G_2 \leftarrow G$. Now pass $G_1$ and $G_2$ as inputs for the subgraph isomorphism sub routine. Let the graph $G = G_2$ have $n$ vertices. This means that if $k > n$ then a clique of size $k$ cannot be a subgraph of $G$. Therefore $k \leq n$. It would take $k^2$ time to make the graph $G_1$ as it has $k$ edges. If $G_1$ is a Subgraph of $G_2$ then G would have a clique of size $k$. Hence if Clique is true then the Subgraph Isomorphism is true and vice versa. Therefore, the Clique can be reduced to Subgraph Isomorphism in polynomial time. Thus, the Subgraph Isomorphism Problem is an NP-Hard Problem

We have shown that the solution certificate of SI is polynomial time verifiable and that SI is NP hard. Hence we can conclude that SI is NP-Complete.

## References


[1] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. *Introduction to Algorithms*. The MIT Press, second edition, 2001.