Assignment #7

1. 34-1: (a),(b)

   (a) The corresponding decision problem is the following: Given $G = [V,E]$, and a positive integer $k$, is there an independence set in $G$ of size $k$ is the decision problem. Thus

   $\text{INDEPENDENCE} = \{(G,k) : G \text{ is a graph with an independence set of size } k\}$

   To show that it is NP, we use the independence set itself as the certificate. It is easy to verify that a given set is indeed an independent set or not and check its size in polynomial time.

   To show that it is NP-complete we show that $\text{CLIQUE} \leq_p \text{INDEPENDENCE}$. Given an instance of CLIQUE with $(G,k)$, let $\overline{G}$ be the complement graph of $G$. Let the corresponding INDEPENDENCE instance be on the $(\overline{G},k)$. There is a clique of size $k$ in graph $G$ iff there is an independent set of size $k$ in $\overline{G}$. In fact the same set does it. Hence the reduction is complete.

   (b) Run INDEPENDENCE over $G$ with values of $k$ in a binary search manner.

2. 34.5-2

   To show that this problem is in NP, we use the vector $x$ as the certificate. We now have to check that it satisfies each constraint in the system $Ax \leq b$ and this can be done in polynomial time. To show that it is NP-complete, we show that $\text{3-SAT} \leq_p 0-1\text{IP}$. To do this for each variable in 3-SAT, we have a variable in 0 – 1IP and for each clause we have a constraint. For example, if a clause looks like $(x_3 \lor \neg x_5 \lor x_8)$ the corresponding constraint would be $y_3 + (1 - y_5) + y_8 \geq 1$. The size of this IP is polynomially related to the boolean expression and the instance of 3-SAT has a yes answer iff the IP has a solution. The reduction is complete.

3. 34.5-5

   To show that HAM-PATH is in NP, we use the path itself as the certificate and it is easy to verify that a given path is hamiltonian in a graph in polynomial time. To show that it is NP-complete, we show the $\text{HAM-CYCLE} \leq_p \text{HAM-PATH}$. (Done in class)

4. 34.1-6: do union, intersection, complementation, and concatenation.

   **Solution:**

   Complementation: Must show that $\{L \in P\} \implies \{\overline{L} \in P\}$

   $L \in P$ implies that there is an algorithm $A$ that decides $L$ and the time taken by $A$ is given by $T(n) = O(n^k)$. This implies that there is a constant $c$ such that $T(n) \leq cn^k$. Now given a string $x \in \{0,1\}^*$ run algorithm
A till either it accepts or rejects $x$. If it accepts then output $x \notin \bar{L}$ else output $x \in \bar{L}$. Thus there is an algorithm $A'$ that decides $\bar{L}$ in polynomial time.

Rest is left to you.

5. 34.2-9

**Solution:**

To do this we must show \( \{L \in P\} \implies \{\bar{L} \in NP\} \).

But \( \{L \in P\} \implies \{\bar{L} \in P\} \) by 34.1-6. Also \( P \subseteq NP \). So the result follows.