Due: July 3, in tutorial[1]

1. Consider the problem **UnaryPartition:**
   Input: $\langle w_1, \ldots, w_n \rangle$ where each $w_i \in \mathbb{N}$ is presented in unary notation.
   Question: Is there a set $S \subseteq \{1, \ldots, n\}$ such that $\sum_{i \in S} w_i = \sum_{i \notin S} w_i$?

   Show that **UnaryPartition** is in P by giving an efficient dynamic programming algorithm that will solve it. Do this as follows:

   (a) Define a suitable array.
   (b) Explain how to solve **UnaryPartition** once your array has been filled in.
   (c) Give a recurrence for the array.

2. Consider the search and decision versions of the problem **Minimum Subset Sum Problem** given as follows:

   MSSP is given by:
   Input: A sequence of weights $\langle w_1, \ldots w_d, t \rangle$, where t is the target amount (all weights and the target are positive integers given in binary).
   Output: A subset $S \subseteq \{1, \ldots, d\}$ such that $\sum_{i \in S} w_i = t$ and $|S|$ is as small as possible, if such a subset exists (return "no such set" otherwise).

   MSSD is given by:
   Input: A sequence of weights $\langle w_1, \ldots w_d, t, B \rangle$, with the bound B (again, all positive integers, presented in binary).
   Question: Is there a subset $S \subseteq \{1, \ldots, d\}$ such that $\sum_{i \in S} w_i = t$ and $|S| \leq B$?

   Show that MSSP $\xrightarrow{p}$ MSSD. Note that to solve MSSP your algorithm should find an optimal set S, not just $|S|$. Also show that MSSD $\in$ NP by giving a "guess-and-check" procedure.

3. Let $G = (U \cup V, E)$ be a bipartite graph with $|U| = |V| = n$. A *perfect matching* in G is a subset $E_0 \subseteq E$ such that $|E_0| = n$ and E_0 is a matching. Let PM be the problem of, given a bipartite graph G, finding a perfect matching; let PMD be the related decision problem that outputs "YES" or "NO" depending on whether a perfect matching exists. Show that PM $\xrightarrow{p}$ PMD. Show that PMD $\in$ NP.

4. Give a Turing machine which recognizes the language $L = \{xx | x \in \{0,1\}^*\}$. Design your Turing machine so that the input alphabet is $\Sigma = \{0,1\}$ and the tape alphabet is $\Gamma = \{0, 1, b\}$.

5. Define the problem **DoublePartition** the same way as Partition, except that we require that $\sum_{i \in S} w_i = 2 \sum_{i \notin S} w_i$. Show that Partition $\leq_p$ DoublePartition.

[1] Assignments are due at the *beginning* of tutorial, and since solutions are discussed in tutorial, late assignments are *not* accepted. The work that you submit *must* be your own. You may discuss problems with each other; however, you should take no notes during these discussions, and you should prepare written solutions alone.