#1

Consider the following graph problem, independent set:

$$\mathsf{INDSET} = \{ \langle G, k \rangle \mid G \text{ has a } k\text{-vertex independent set} \}.$$

An independent set is a subset of the vertices of a graph where no pair of vertices in the independent set have an edge between them. Like the previous lab, let's go through a systematic process for proving this claim. The only required parts (that you must turn in) are part 2 and part 6.

1. First, **explore the problem space**. Come up with two examples of instances of $\langle G, k \rangle$, one that is in INDSET and one that is not in the language.
2. Next, **show that INDSET $\in$ NP.**
3. Now, **identify the inputs** to CLIQUE and INDSET.
4. With this, **identify the inputs and outputs of the reduction function** we must design.
5. Now, onto exploration of designs. In the previous lab, we essentially "forced" PUZZLE to solve 3SAT. However, this perspective on designing the reduction will not be as fruitful with these problems even though they operate on the same domain. To make forward progress, I encourage you to instead consider how CLIQUE and INDSET are actually the same problem. More concretely, I recommend looking at the definition of clique and independent set and analyzing their relationship with each other.

   Just like the previous lab, **come up with at least two potential designs** for the reduction function and **push them through on a toy CLIQUE instance**. For each design that does not work, describe what was broken and how you could potentially fix it.

   (Hint: it might be useful to think about graph complements for this reduction.)
6. Finally, once you have a plausible reduction strategy, formally prove that CLIQUE $\leq_p$ INDSET.

#2 More Graph Reductions (Optional)

Here is another NP-completeness problem for you to consider, following the trajectory of Karp and his 21 NP-complete problems[1]:

$$\mathsf{SETPACK} = \{ \langle \mathcal{U}, S, k \rangle \mid S \text{ is a family of subsets of } \mathcal{U}, \text{ there is a set packing of } S \text{ of size at least } k. \}.$$

A set packing of size k is a family S of subsets of $\mathcal{U}$ with $|U| \geq k$ that are pairwise disjoint, i.e., no sets share an element in common. Clearly, the empty set is a valid set packing for any set—instead, we are interested in maximizing the size of the set packing while obeying the restriction that no pairs of sets in the packing share a common element.

As with the previous lab questions, systematically work through the process of showing that SETPACK is indeed NP-complete. In your deliberations, you should plan to reduce the INDSET problem to SETPACK.

(Hint: here, the first perspective on reductions will be more useful. Force SETPACK to solve a INDSET instance for you.)

[1] Karp R.M. (1972) Reducibility among Combinatorial Problems. In: Miller R.E., Thatcher J.W., Bohlinger J.D. (eds) Complexity of Computer Computations. The IBM Research Symposia Series. Springer, Boston, MA.