Note: Your TA probably will not cover all the problems. This is totally fine, the discussion worksheets are not designed to be finished in an hour. They are deliberately made long so they can serve as a resource you can use to practice, reinforce, and build upon concepts discussed in lecture, readings, and the homework.

1 NP or not NP, that is the question

For the following questions, circle the (unique) condition that would make the statement true.

(a) If B is **NP**-complete, then for any problem $A \in \mathbf{NP}$, there exists a polynomial-time reduction from A to B.

Always True   True iff $\mathbf{P} = \mathbf{NP}$ True iff $\mathbf{P} \neq \mathbf{NP}$ Always False

(b) If B is in **NP**, then for any problem $A \in \mathbf{P}$, there exists a polynomial-time reduction from A to B.

Always True   True iff $\mathbf{P} = \mathbf{NP}$ True iff $\mathbf{P} \neq \mathbf{NP}$ Always False

(c) 2 SAT is **NP**-complete.

Always True   True iff $\mathbf{P} = \mathbf{NP}$ True iff $\mathbf{P} \neq \mathbf{NP}$ Always False

(d) Minimum Spanning Tree is in **NP**.

Always True   True iff $\mathbf{P} = \mathbf{NP}$ True iff $\mathbf{P} \neq \mathbf{NP}$ Always False

2 Graph Coloring Problem

In the k-coloring problem, we are given an undirected graph $G = (V, E)$ and are asked to assign every vertex a color from the set $1, \cdots, k$, such that no two adjacent vertices have the same color.
As you will prove in the homework 3-coloring is NP-Complete.

Prove that 10-coloring is also NP-Complete.

3 2-SAT and Variants

Max-2-SAT is defined as follows. Let $C_1, \ldots, C_m$ be a collection of 2-clauses and b a non-negative integer. We want to determine if there is some assignment which satisfies at least b clauses.

Max-Cut is defined as follows. Let G be an undirected unweighted graph, and k a non-negative integer. We want to determine if there is some cut with at least k edges crossing it. Max-Cut is known to be NP-complete.

Show that Max-2-SAT is NP-complete by reducing from Max-Cut. Prove the correctness of your reduction.

4 (3,3)-SAT

Consider the (3,3)-SAT problem, which is the same as 3-SAT except each literal or its negation appears *at most* 3 times across the entire formula. Notice that (3,3)-SAT is reducible to 3-SAT because every formula that satisfies the (3,3)-SAT constraints satisfies those for 3-SAT. We are interested in the other direction.

Show that (3,3)-SAT is NP-Hard via reduction from 3-SAT. By doing so, we will have eliminated the notion that the "hardness" of 3-SAT was in the repetition of variables across the formula.

Give a precise description of the reduction and prove its correctness.