Name: Vera Duong

ID: 109431166

CSCI 3104, Algorithms **Due THURSDAY, APRIL 29, 2021**

Problem Set 10 (50 points) **Spring 2021, CU-Boulder**

Advice 1: For every problem in this class, you must justify your answer: show how you arrived at it and why it is correct. If there are assumptions you need to make along the way, state those clearly.
Advice 2: Verbal reasoning is typically insufficient for full credit. Instead, write a logical argument, in the style of a mathematical proof.

Instructions for submitting your solution:

- The solutions **should be typed** and we cannot accept hand-written solutions. Here's a short intro to Latex.
- You should submit your work through **Gradescope** only.
- The easiest way to access Gradescope is through our Canvas page. There is a Gradescope button in the left menu.
- Gradescope will only accept **.pdf** files.
- It is vital that you match each problem part with your work. Skip to 1:40 to just see the matching info.

Name: Vera Duong

ID: 109431166

CSCI 3104, Algorithms **Due THURSDAY, APRIL 29, 2021**

Problem Set 10 (50 points) **Spring 2021, CU-Boulder**

1. Let P_1, P_2 be two problems such that $P_1 \leq_p P_2$. That is, we have a polynomial-time reduction $r : P_1 \to P_2$. If we assume $P_2 \in P$, explain why this implies that $P_1 \in P$.

   SOLUTION:

   We are given that $P_1 \leq_p P_2$, meaning P_2 is at least as hard as P_1, or in other words P_1 is at most as hard as P_2. We also have a polynomial-time reduction $r : P_1 \to P_2$. If there is a polynomial time algorithm solving P_2, then this means that there is a polynomial time algorithm solving P_1. We are assuming that $P_2 \in P$, so based on the statement earlier, the polynomial time algorithm solving P_2 will also solve P_1. Hence, this implies that $P_1 \in P$.

Name: Vera Duong
ID: 109431166

CSCI 3104, Algorithms **Due THURSDAY, APRIL 29, 2021**
Problem Set 10 (50 points) **Spring 2021, CU-Boulder**

2. Recall the k-Colorability problem.

   - Input: Let G be a simple, undirected graph.
   - Decision: Can we color the vertices of G using exactly k colors, such that whenever u and v are adjacent vertices, u and v receive different colors?

   It is well known that k-Colorability belongs to NP, and 3-Colorability is NP-complete. Our goal is to show that the 4-Colorability problem is also NP-complete. To do so, we reduce the 3-Colorability problem to the 4-Colorability problem. We provide the reduction below. The questions following the reduction will ask you to verify its correctness.

   Reduction: Let G be a simple, undirected graph. We construct a new simple, undirected graph H by starting with a copy of G. We then add a new vertex t to H, and for each vertex $v \in V(G)$ we add the edge tv to $E(H)$.

   Your job is to verify the correctness of the reduction by completing parts (a)–(c) below, which completes the proof that 4-Colorability is NP-complete. For the rest of this problem, let G be a graph, and let H be the result of applying the reduction to G.

   (a) Suppose that G is colorable using exactly 3 colors. Argue that H is colorable using exactly 4 colors.

   (b) Suppose that H is colorable using exactly 4 colors. Argue that G is colorable using exactly 3 colors.

   (c) Let n be the number of vertices in G. Carefully explain why H can be constructed in time polynomial in n. [**Hint:** Count the number of vertices and edges we add to G in order to obtain H.]

   SOLUTION:

   (a) When adding the new vertex t in the constructed graph H, we can color t to be the 4^{th} color while keeping the "original" vertices from G the same 3 colors. In this way, we will satisfy the statement that H is colorable using exactly 4 colors.

   (b) We know that H contains an extra vertex t that is connected to all the other vertices (the same ones in G). By the colorability problem, this implies that all the vertices in H excluding t must be a different color than t because all the other vertices have an edge connected to t. The vertices in H (excluding t) must also be colored with exactly 3 colors since t has to be the 4^{th} color. We know that t is not included in G, but it's included in H, so this satisfies the statement that G is colorable using exactly 3 colors.

   (c) Counting the number of vertices and edges we add to G in order to obtain H, we would need to add one vertex t and n edges. This implies that it takes linear time to add a single vertex and n edges to construct H from G with cost of $n + 1$. This in turn is time polynomial in n.