

Prove that each of the following problems is NP-hard.

1. Given a graph $(G = (V, E))$ and a positive integer k , is there a subset $V' \subseteq V$ such that $|V'| \leq k$ that deleting the vertices of V' from G leaves a DAG.

Solution: HW Problem ■

2. Given an undirected graph G , does G contain a simple path that visits all but 374 vertices?

Solution: We prove this problem is NP-hard by a reduction from the undirected Hamiltonian path problem. Given an arbitrary graph G , let H be the graph obtained from G by adding 374 isolated vertices. Call a path in H **almost-Hamiltonian** if it visits all but 374 vertices. I claim that G contains a Hamiltonian path if and only if H contains an almost-Hamiltonian path.

- \Rightarrow Suppose G has a Hamiltonian path P . Then P is an almost-Hamiltonian path in H , because it misses only the 374 isolated vertices.
- \Leftarrow Suppose H has an almost-Hamiltonian path P . This path must miss all 374 isolated vertices in H , and therefore must visit every vertex in G . Every edge in H , and therefore every edge in P , is also an edge in G . We conclude that P is a Hamiltonian path in G .

Given G , we can easily build H in polynomial time by brute force. ■

3. Given an undirected graph G , does G have a spanning tree in which every node has degree at most 374?

Solution: We prove this problem is NP-hard by a reduction from the undirected Hamiltonian path problem. Given an arbitrary graph G , let H be the graph obtained by attaching a fan of 372 edges to every vertex of G . Call a spanning tree of H **almost-Hamiltonian** if it has maximum degree 374. I claim that G contains a Hamiltonian path if and only if H contains an almost-Hamiltonian spanning tree.

- \Rightarrow Suppose G has a Hamiltonian path P . Let T be the spanning tree of H obtained by adding every fan edge in H to P . Every vertex v of H is either a leaf of T or a vertex of P . If $v \in P$, then $\deg_P(v) \leq 2$, and therefore $\deg_H(v) = \deg_P(v) + 372 \leq 374$. We conclude that T is an almost-Hamiltonian spanning tree.
- \Leftarrow Suppose H has an almost-Hamiltonian spanning tree T . The leaves of T are precisely the vertices of H with degree 1; these are also precisely the vertices of H that are not vertices of G . Let P be the subtree of T obtained by deleting every leaf of T . Observe that P is a spanning tree of G , and for every vertex $v \in P$, we have $\deg_P(v) = \deg_T(v) - 372 \leq 2$. We conclude that P is a Hamiltonian path in G .

Given G , we can easily build H in polynomial time by brute force. ■

4. Given an undirected graph G , does G have a spanning tree with at most 374 leaves?

Solution: We prove this problem is NP-hard by a reduction from the undirected Hamiltonian path problem.^a Given an arbitrary graph G , let H be the graph obtained from G by adding the following vertices and edges:

- First we add a vertex z with edges to every other vertex in z .
- Then we add 373 vertices $\ell_1, \dots, \ell_{373}$, each with edges to t and nothing else.

Call a spanning tree of H **almost-Hamiltonian** if it has at most 374 leaves. I claim that G contains a Hamiltonian path if and only if H contains an almost-Hamiltonian spanning tree.

- \Rightarrow Suppose G has a Hamiltonian path P . Suppose P starts at vertex s and ends at vertex t . Let T be subgraph of H obtained by adding the edge tz and all possible edges $z\ell_i$. Then T is a spanning tree of H with exactly 374 leaves, namely s and all 373 new vertices ℓ_i .
- \Leftarrow Suppose H has an almost-Hamiltonian spanning tree T . Every node ℓ_i is a leaf of T , so T must consist of the 373 edges $z\ell_i$ and a simple path from z to some vertex s of G . Let t be the only neighbor of z in T that is not a leaf ℓ_i , and let P be the unique path in T from s to t . This path visits every vertex of G ; in other words, P is a Hamiltonian path in G .

Given G , we can easily build H in polynomial time by brute force. ■

^aAre you noticing a pattern here?

5. Recall that a 5-coloring of a graph G is a function that assigns each vertex of G a “color” from the set $\{0, 1, 2, 3, 4\}$, such that for any edge uv , vertices u and v are assigned different “colors”. A 5-coloring is **careful** if the colors assigned to adjacent vertices are not only distinct, but differ by more than 1 (mod 5). Prove that deciding whether a given graph has a careful 5-coloring is NP-hard.

Solution: We prove that careful 5-coloring is NP-hard by reduction from the standard 5COLOR problem.

Given a graph G , we construct a new graph H by replacing each edge in G with a path of length three. I claim that H has a careful 5-coloring if and only if G has a (not necessarily careful) 5-coloring.

\Leftarrow Suppose G has a 5-coloring. Consider a single edge uv in G , and suppose $color(u) = a$ and $color(v) = b$. We color the path from u to v in H as follows:

- If $b = (a + 1) \bmod 5$, use colors $(a, (a + 2) \bmod 5, (a - 1) \bmod 5, b)$.
- If $b = (a - 1) \bmod 5$, use colors $(a, (a - 2) \bmod 5, (a + 1) \bmod 5, b)$.
- Otherwise, use colors (a, b, a, b) .

In particular, every vertex in G retains its color in H . The resulting 5-coloring of H is careful.

\Rightarrow On the other hand, suppose H has a careful 5-coloring. Consider a path (u, x, y, v) in H corresponding to an arbitrary edge uv in G . There are exactly eight careful colorings of this path with $color(u) = 0$, namely: $(0, 2, 0, 2)$, $(0, 2, 0, 3)$, $(0, 2, 4, 1)$, $(0, 2, 4, 2)$, $(0, 3, 0, 3)$, $(0, 3, 0, 2)$, $(0, 3, 1, 3)$, $(0, 3, 1, 4)$. It follows immediately that $color(u) \neq color(v)$. Thus, if we color each vertex of G with its color in H , we obtain a valid 5-coloring of G .

Given G , we can clearly construct H in polynomial time. ■

6. Prove that the following problem is NP-hard: Given an undirected graph G , find *any* integer $k > 374$ such that G has a proper coloring with k colors but G does not have a proper coloring with $k - 374$ colors.

Solution: Let G' be the union of 374 copies of G , with additional edges between *every* vertex of each copy and *every* vertex in *every* other copy. Given G , we can easily build G' in polynomial time by brute force. Let $\chi(G)$ and $\chi(G')$ denote the minimum number of colors in any proper coloring of G , and define $\chi(G')$ similarly.

\Rightarrow Fix any coloring of G with $\chi(G)$ colors. We can obtain a proper coloring of G' with $374 \cdot \chi(G)$ colors, by using a distinct set of $\chi(G)$ colors in each copy of G . Thus, $\chi(G') \leq 374 \cdot \chi(G)$.

\Leftarrow Now fix any coloring of G' with $\chi(G')$ colors. Each copy of G in G' must use its own distinct set of colors, so at least one copy of G uses at most $\lfloor \chi(G')/374 \rfloor$ colors. Thus, $\chi(G) \leq \lfloor \chi(G')/374 \rfloor$.

These two observations immediately imply that $\chi(G') = 374 \cdot \chi(G)$. It follows that if k is an integer such that $k - 374 < \chi(G') \leq k$, then $\chi(G) = \chi(G')/374 = \lceil k/374 \rceil$. Thus, if we could compute such an integer k in polynomial time, we could compute $\chi(G)$ in polynomial time. But computing $\chi(G)$ is NP-hard! ■

7. A **bicoloring** of an undirected graph assigns each vertex a set of *two* colors. There are two types of bicoloring: In a *weak* bicoloring, the endpoints of each edge must use *different* sets of colors; however, these two sets may share one color. In a *strong* bicoloring, the endpoints of each edge must use *distinct* sets of colors; that is, they must use four colors altogether. Every strong bicoloring is also a weak bicoloring.
- (a) Prove that finding the minimum number of colors in a weak bicoloring of a given graph is NP-hard.

Solution: It suffices to prove that deciding whether a graph has a weak bicoloring with three colors is NP-hard, using the following trivial reduction from the standard 3COLOR problem.

Let G be an arbitrary undirected graph. I claim that G has a proper 3-coloring if and only if G has a weak bicoloring with 3 colors.

- Suppose G has a proper coloring using the colors red, green, and blue. We can obtain a weak bicoloring of G using only the colors cyan, magenta, and yellow by recoloring each red vertex with {magenta, yellow}, recoloring each blue vertex with {magenta, cyan}, and recoloring each green vertex with {yellow, cyan}.
- Suppose G has a weak bicoloring using the colors cyan, magenta, yellow. Then we can obtain a proper 3-coloring of G by defining red = {magenta, yellow}, defining blue = {magenta, cyan}, and defining green = {yellow, cyan}.

More generally, for any integer k and any graph G , every weak k -bicoloring of G is also a proper $\binom{k}{2}$ -coloring of G , and vice versa. ■

- (b) Prove that finding the minimum number of colors in a strong bicoloring of a given graph is NP-hard.

Solution: It suffices to prove that deciding whether a graph has a strong bicoloring with six colors is NP-hard, using the following reduction from the standard 3Color problem.

Let G be an arbitrary undirected graph. We build a new graph H from G as follows:

- For every vertex v in G , the graph H contains three vertices v_1 , v_2 , and v_3 and three edges v_1v_2 , v_2v_3 , and v_3v_1 .
- For every edge uv in G , the graph H contains three edges u_1v_1 , u_2v_2 , and u_3v_3 .

I claim that G has a proper 3-coloring if and only if H has a strong bicoloring with six colors. Without loss of generality, we can assume that G (and therefore H) is connected; otherwise, consider each component independently.

⇒ Suppose G has a proper 3-coloring with colors red, green, and blue. Then we define a strong bicoloring of H with colors 1, 2, 3, 4, 5, 6 as follows:

- For every red vertex v in G , let $color(v_1) = \{1, 2\}$ and $color(v_2) = \{3, 4\}$ and $color(v_3) = \{5, 6\}$.
- For every blue vertex v in G , let $color(v_1) = \{3, 4\}$ and $color(v_2) = \{5, 6\}$ and $color(v_3) = \{1, 2\}$.
- For every green vertex v in G , let $color(v_1) = \{5, 6\}$ and $color(v_2) = \{1, 2\}$ and $color(v_3) = \{3, 4\}$.

Exhaustive case analysis confirms that every pair of adjacent vertices of H has disjoint color sets.

- Suppose H has a strong bicoloring with six colors. Fix an arbitrary vertex v in G , and without loss of generality, suppose $color(v_1) = \{1, 2\}$ and $color(v_2) = \{3, 4\}$ and $color(v_3) = \{5, 6\}$. Exhaustive case analysis implies that for any edge uv , each vertex u_i must be colored either $\{1, 2\}$ or $\{3, 4\}$ or $\{5, 6\}$. It follows by induction that every vertex in H must be colored either $\{1, 2\}$ or $\{3, 4\}$ or $\{5, 6\}$.

Now for each vertex w in G , color w red if $color(w_1) = \{1, 2\}$, blue if $color(w_2) = \{3, 4\}$, and green if $color(w_3) = \{5, 6\}$. This assignment of colors is a proper 3-coloring of G .

Given G , we can build H in polynomial time by brute force. ■

I believe that deciding whether a graph has a strong bicoloring with five colors is also NP-hard, but I don't have a proof yet. A graph has a strong bicoloring with four colors if and only if it is bipartite, and a strong bicoloring with two or three colors if and only if it has no edges.