Solution 8

1. Suppose $2n$ teams play in a round-robin tournament. Over a period of $2n-1$ days, every team plays every other team exactly once. There are no ties. Show that for each day we can select a winning team, without selecting the same team twice. (Hint: Consider a bipartite graph with a set of team vertices and a set of day vertices. Take any set of days W and assume not all teams won in some day in W. Let t_w be a team that did not win in any day in W. Consider the implication on the number of teams that won at least once in some day in W. Invoke Hall's Theorem.)

   Solution:

   We construct a bipartitie graph $G = (V, E)$, where each day and each team is represented as a vertex in G. A team vertex t_i is connected to a day vertex d_j if team i wins the match on day j. The problem is equivalent to there exists a maximal matching with size $2n-1$. Let the set of team vertices be T and set of day vertices be D. By Hall's theorem, it is sufficient to show for any subset $S \subseteq D$, the neighbour vertices of S in T has size at least $|S|$. In other words, for any set of k days, wehre $1 \leq k \leq 2n-1$, there are at least k distinct teams wins at least one match in k days. By way of contradiction, if not, there exists a set of k days, such that there are less than k distinct teams won at least one match. There must exist a team t that lose every game in these k days, otherwise there are $2n$ distinct teams won a a match. However, since team t lose every game in these k days, there are at least k different winners, which is a contradiction.

2. Given an undirected graph $G = (V, E)$ and an integer k. A clique of G is a subset $V' \subseteq V$ of vertices, each pair of which is connected by an edge in E. The **Clique** problem asks whether G contains a clique of size at least k. An independent set of G is a subset $V' \subseteq V$ of vertices such that each edge in E is incident on at most one vertex in V'. The **Independent-Set** problem asks whether G' contains an independent set of size at least k'. We proved in class that the Clique problem is *NP*-complete. Show that the independent set problem is same by reduction from Clique problem.

   Show a polynomial reduction from **Clique** problem to **Independent-Set** problem.

   Solution:

   Given a instance of Clique problem $G = (V, E)$ and k. We construct an instance of Independent-Set $G' = (V, E')$ and $k' = k$, where G' is complement graph of G, i.e., $V' = V$ and $v_i - v_j$ appears in E' if $v_i - v_j$ is not in E for all i and j. Obviously, the construction takes polynomial time.

   Theorem 1. *G contains a clique with size of k if and only if G' contains an independent set of size k.*

   Proof. Let $C \subseteq V$ be the clique in G. It is not difficult to see that C is an independent set of size k in G' because any two vertices in C don't have an edge between them according to our construction. The necessity is due to the same reason. □

3. Show that the following three problems are polynomial time reducible to each other.

   - **Set-Cover**: Given a collection of sets, and a number k, the Set-Cover problem asks if there are at most k sets from the collection of sets such that their union contains every element in the union of all sets.
   - **Hitting-Set**: Given a collection of sets, and a number k, the Hitting-Set problem asks if are there at most k elements of the sets such that there is at least one element from each set?

- **Dominating-Set**: Given an undirected graph G, and a number k, the Dominating-Set problem asks if there is a subset of vertices of size $\leq k$ such that every vertex in the graph is either in the subset or has a neighbor that is in the subset.

Prove Set-Cover, Hitting-Set and Dominating-Set are polynomial-time reducible to each other.

(Hint: One strategy is to show Set-Cover $\leq_p$ Hitting-Set, Hitting-Set $\leq_p$ Dominating-Set and Dominating-Set $\leq_p$ Set-Cover. An alternative way is to show Hitting-Set $\leq_p$ Dominating-Set, Dominating-Set $\leq_p$ Hitting-Set, Set-Cover $\leq_p$ Dominating-Set and Dominating-Set $\leq_p$ Set-Cover. In class we have seen Vertex-Cover reduced poly to Dominating-Set).

Solution:

(a) **Hitting-Set $\leq_p$ Dominating-Set.**

Consider the following construction from an instance of Hitting Set to an instance of Dominating Set:

> We construct the Graph G as follows:
>
> i. For each set and element, create a node that represent it.
>
> ii. For each set, make a clique for the nodes i.e., connect all the elements node and the set node they belong to
>
> iii. $k' = k$

Poof:

$\Rightarrow$: If there exists a Hitting set with size k, we select the nodes that represents the k elements. These k nodes is dominating set for G since every set nodes is connected to at least one of the k nodes by the Hitting Set property. All the elements in the same set is a clique so all nodes is either in the dominating set or is a neighbor of the dominating set.

$\Leftarrow$: If there exists a dominating set with size k, we first change the set that only contains element nodes : for each set node in the dominating set we select an arbitrary element node it connected to. Since a set nodes and its elements node is clique, the above change results a dominate set for G and it only contains element nodes. Now for each element node, we select the element represented by the node and get a hitting set with the same size. Since every set node is connected to some element nodes, by the correspondence of our construction the hitting set contains at least one element for each set.

(b) **Dominating-Set $\leq_p$ Hitting-Set.**

Consider the following construction from Dominating Set to an instance of Hitting Set:

> We construct a collection of sets as follows:
>
> i. For each node i, create a set S_i.
>
> ii. For each set S_i, we add node i and its neighbors
>
> iii. $k' = k$

Poof:

$\Rightarrow$: If there is a dominating set with size k, we can construct a hitting set with size k by taking the elements corresponds these nodes. For each set S_i, either element i is in the hitting set or an element in S_i is in the hitting set since every node i is either in the dominating set or it is the neighbor of some node in the dominating set.

$\Leftarrow$: If there is a hitting set with size k, then by select the nodes that represent the elements in the hitting set, we get a set with size k. The set is dominating set since every set has has an element in the hitting set and so every node is either in the dominating set or has a neighbor in the dominating set because of our construction.

(c) **Set-Cover $\leq_p$ Dominating-Set**.

Consider the following construction from an instance of Set Cover to an instance of Dominating Set:

> We construct the Graph G as follows:
>
> i. For each set and element, create a node that represent it.
>
> ii. For each set S, connect each set node with the elements node in the set S
>
> iii. Make a clique for all set nodes.
>
> iv. $k' = k$

Poof:

$\Rightarrow$: If there exists a set cover with size k, we select the set nodes that represents the k sets. These k set nodes is dominating set for G since every set nodes is connected to the elements node that in the set. Since the union of set cover is the universe, the set nodes we select connects with all element nodes. Moreover, all the set nodes constitute a clique. Hence, every node in the graph G is either selected or is a neighbor of a node we select.

$\Leftarrow$: If there exists a dominating set with size k, we first change the set that only contains set nodes : for each element node i in the dominating set we select an arbitrary set node j it connected to. It is easy to see that the nodes j connects to all set nodes and node i. Since node j only connect to set nodes, the new set is also a dominating set with the same size. Now for each set node in the dominating set, we select the set represented by the node and get a set cover with the same size. This is because every element nodes connect to some set node in the dominating set, which means every element belongs to some set in the set cover.

(d) **Dominating-Set $\leq_p$ Set-Cover**.

The construction is exactly the same as problem (b) except that we are selecting a subset of sets instead of elements. The proof is also almost the same.

4. Given a directed graph $G = (V, E)$ and a pair of vertices s, t in G, the **Hamiltonian Path** problem asks whether there is a simple path from s to t that visits every vertex of G exactly once. The **Hamiltonian Cycle** problem asks if there is a cycle in a directed graph G that visits every vertex exactly once. Show that Hamiltonian Path and Hamiltonian Cycle problems are polynomial-time reducible to each other.

Solution:

`HamiltonianPath` $\leq_p$ `HamiltonianCycle`

Given an instance of `HamiltonianPath` problem (G, s, t), we create a new graph G': Add a new vertex v and two edge $\{t, v\}, \{v, s\}$ to G.

It's easy to see that there is a Hamiltonian path from s to t in G if and only if there is a Hamiltonian cycle in G'.

$\texttt{HamiltonianCycle} \leq_p \texttt{HamiltonianPath}$

Given an instance of `HamiltonianCycle` G, we create a instance of `HamiltonianPath` G': Let G' be copy of G; add two new vertex s, t. For every node v in G that has an edge $\{s, v\}$, add an edge $\{v, t\}$ in G'.

Again it can easily be shown that there is a Hamiltonian cycle in G if and only if there is a Hamiltonian path from s to t in G'.