COL351
Assignment 4

Mallika Prabhakar (2019CS50440)
Sayam Sethi (2019CS10399)

September 2021

Contents

1 Question 1

Question 1

Question.

Solution. $\square$

2 Question 2

2.1 2.1

Question 2.1

Question. *Consider a set $U = u_1, ..., u_n$ of n elements and a collection $A_1, A_2, ..., A_m$ of subsets of U. That is, $A_i \subseteq U$, for $i \in [1, m]$. We say that a set $S \subseteq U$ is a hitting-set for the collection $A_1, A_2, ..., A_m$ if $S \cap A_i$ is non-empty for each i. The Hitting-Set Problem (HS) for the input $(U, A_1, ..., A_m)$ is to decide if there exists a hitting set $S \subseteq U$ of size at most k. Prove that Hitting-Set problem is in NP class.*

Solution.

Claim 2.1. *Hitting-Set Problem is a decision problem.*

Proof. For hitting-set problem, we want to observe if the given set of subsets is a hitting set or not. Since the answer is a boolean yes or no, it is a decision problem. □

Claim 2.2. *Hitting-set problem can be verified in polynomial time*

Proof. Given hitting-set S, to verify if it is a solution or not, we perform the following checks:

1. $|S| \leq k$
2. $S \cap A_i$ for all $A_i \in$ set $A_1, A_2, ..., A_m$ is $\neq \phi$

Check 1 is a constant order operation.
For check 2, in worst case scenario, $|A_i| = n \; \forall \; i$. Time to find intersection of all $A_i s$ and S is $O(k \times n)$. Total time taken is $O(k \times n \times m)$
Total time taken for verification hence is $O(1) + O(nmk) = O(nmk)$ which is polynomial time. □

Claim 2.3. *A decision problem is an NP Class problem if it has a polynomial time verifier.*

Proof. As shown in 2.1 and 2.2, Hitting set problem is a decision problem with a polynomial time verifier. Hence it is NP Class. □

□

2.2 2.2

Question 2.2

Question. *Prove that Hitting Set is NP-complete by reducing Vertex-cover to Hitting Set.*

Solution. □

3 Question 3

3.1 3.1

Question 3.1

Question. *Prove that Undirected Feedback Set Problem is in NP class.*

Proof. The algorithm for checking if a given set X of size k is a valid feedback set is as follows:

Algorithm 1 Verifier for UFS

```
procedure VerifyUFS(G, X, k)
    H ← G \ X
    return |X| ≤ k ∧ HasCycle(H)
end procedure
```

We know that checking for a cycle can be done using DFS or BFS in time linear to the size of the graph ($O(n + m)$ time). We have also proved the correctness of the algorithm in the earlier part of the course. Therefore, we have shown that the UFS problem is in NP class since the verifier has a polynomial time algorithm. □

3.2 3.2

Question 3.2

Question. *Prove that Undirected Feedback Set Problem is NP-complete by reducing Vertex-cover to Undirected Feedback Set Problem.*

Proof. We will transform a given instance of vertex-cover problem to UFS as follows:

Algorithm 2 Reducing Vertex-cover to UFS

```
procedure ReduceVC(G, k)
    G' ← emptyGraph()
    for e = {u, v} ∈ E(G) do
        G' ← addEdge(G, {u, x_uv})
        G' ← addEdge(G, {u, y_uv})
        G' ← addEdge(G, {v, x_uv})
        G' ← addEdge(G, {u, y_uv})
    end for
    return G', k
end procedure
```

Thus, we transform an instance of the vertex cover problem of G, k to an instance of UFS problem of G', k. We notice that the transformation involves adding 4 edges to G' for each edge in G. Therefore the time complexity is linear in the size of G. We now show the correctness of the transformation:

Claim 3.1. *Every 'yes' instance of $VC(G,k)$ has a 'yes' instance of $UFS(G',k)$ and vice versa*

Proof. ($\Longrightarrow$) Consider any set X which is a Vertex-cover of size atmost k in G. We will now show that this set X is also a UFS of size atmost k. Since X is a vertex cover, then for every edge $e = \{u, v\}$ in G, atleast one of u, v is in X. Therefore, for eacy cycle of size 4 (there are only cycles of length 4 and no other cycles) in G', atleast one of the vertex has been removed. Therefore there is no more cycle remaining in G'. Thus, we have shown that every 'yes' instance of $VC(G,k)$ has been mapped to a 'yes' instance of $UFS(G',k)$ by using the same solution.
($\Longleftarrow$) Consider any set Y which is a UFS of G' with size atmost k. Now, we will generate a set Y_m such that it contains no vertex of the kind x_{uv} or y_{uv} and has a cardinality atmost $|Y|$. To do this, we observe that for each cycle, there is atleast one vertex out of u, v, x_{uv}, y_{uv} present in Y. If either of x_{uv}, y_{uv} is present in Y, we define Y_1 as $Y \setminus \{w\} \cup \{u\}$ else we define $Y_1 = Y$. Therefore, Y_1 has a size $\leq |Y|$. We proceed similarly for each edge $e \in E(G)$ until we obtain $Y_{|E|} = Y_m$. Now, Y_m is also a valid solution of $UFS(G',k)$ since none of the edges x_{uv}, y_{uv} appear in any other cycle and the size of $Y_m \leq Y \leq k$. Additionally, all the vertices present in Y_m are also present in $V(G)$. We now show that Y_m is a vertex-cover of G. This is easy to see since atleast one of u, v is present in Y_m for each edge $e \in E(G)$. Therefore we have shown that any 'yes' instance of $UFS(G',k)$ can be mapped to a 'yes' instance of $VC(G,k)$.
Therefore we have shown both sides of the proof and this completes the proof of the claim. □

Now, we have shown a mapping for every problem of $VC(G,k)$ to $UFS(G',k)$ and have also shown that the 'yes' instances of both problems are equivalent.Therefore, we have shown that $VC \leq_p UFS$ and since we know from lectures that VC is NP-complete, UFS is also an NP-complete problem. This completes the proof of showing NP-completeness of UFS by reducing Vertex-cover to UFS. □