COL351

Assignment 4

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

${\bf September}\ 2021$

Contents

1	Question 1	2
	Question 2	3
	2.1 2.1	3
	2.2 2.2	3
3	Question 3	4
	3.1 3.1	4
	3.2 3.2	4

1 Question 1



2 Question 2

2.1 2.1

Question 2.1	
subsets of U . That the collection A_1 , A for the input (U, A_1)	der a set $U = u_1,, u_n$ of n elements and a collection $A_1, A_2,, A_m$ of t is, $A_i \subseteq U$, for $i \in [1, m]$. We say that a set $S \subseteq U$ is a hitting-set for $u_2,, u_m$ if $S \cap A_i$ is non-empty for each i . The Hitting-Set Problem (HS) $u_1,, u_m$ is to decide if there exists a hitting set $S \subseteq U$ of size at most k . Set problem is in NP class.
Solution.	
Claim 2.1. Hittin	g-Set Problem is a decision problem.
-	set problem, we want to observe if the given set of subsets is a hitting set nswer is a boolean yes or no, it is a decision problem. \Box
Claim 2.2. Hittin	g-set problem can be verified in polynomial time
Proof. Given hittin	g-set S , to verify if it is a solution or not, we perform the following checks:
$1. S \le k$	
2. $S \cap A_i$ for all	$A_i \in \text{set } A_1, A_2,, A_m \text{ is } \neq \phi$
For check 2, in wor is $O(k \times n)$. Total	ant order operation. The section of all A_i and S_i and S_i time taken is $O(k \times n \times m)$ for verification hence is $O(1) + O(nmk) = O(nmk)$ which is polynomial \square
Claim 2.3. A deca	ision problem is an NP Class problem if it has a polynomial time verifier.
<i>Proof.</i> As shown in time verifier. Hence	a 2.1 and 2.2, Hitting set problem is a decision problem with a polynomial e it is NP Class. $\hfill\Box$

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

```
 \begin{aligned} & \textbf{procedure VerifyUFS}(G, X, k) \\ & H \leftarrow G \setminus X \\ & \textbf{return } |X| \leq k \wedge HasCycle(H) \\ & \textbf{end procedure} \end{aligned}
```

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

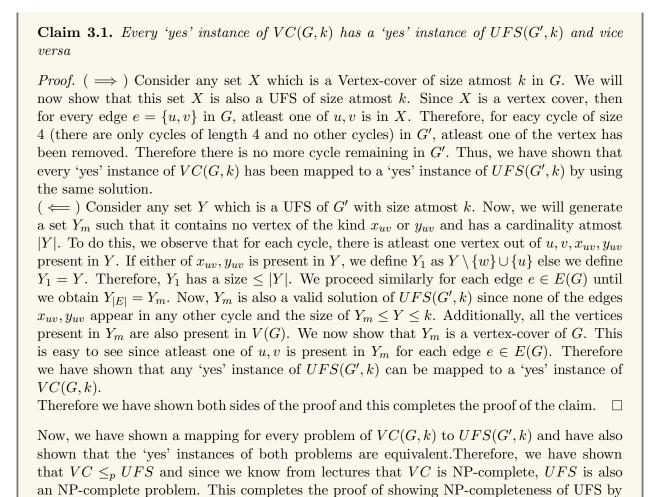
Question. Prove that Undirected Feedback Set Problem is NP-complete by reducing Vertex-cover to Undirected Feedback Set Problem.

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' \leftarrow emptyGraph()
for e = \{u, v\} \in E(G) do
G' \leftarrow addEdge(G, \{u, x_{uv}\})
G' \leftarrow addEdge(G, \{u, y_{uv}\})
G' \leftarrow addEdge(G, \{v, x_{uv}\})
G' \leftarrow 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:



reducing Vertex-cover to UFS.