

COMP 3804 — Assignment 4

Due: Thursday April 6, 23:59.

Assignment Policy:

- Your assignment must be submitted as one single PDF file through Brightspace.

Use the following format to name your file:

LastName_StudentId_a4.pdf

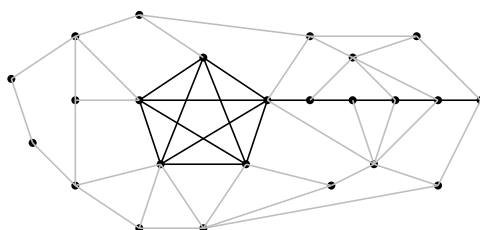
- **Late assignments will not be accepted. I will not reply to emails of the type “my internet connection broke down at 23:57” or “my scanner stopped working at 23:58”, or “my dog ate my laptop charger”.**
- You are encouraged to collaborate on assignments, but at the level of discussion only. When writing your solutions, you must do so in your own words.
- Past experience has shown conclusively that those who do not put adequate effort into the assignments do not learn the material and have a probability near 1 of doing poorly on the exams.
- When writing your solutions, you must follow the guidelines below.
 - You must justify your answers.
 - The answers should be concise, clear and neat.
 - When presenting proofs, every step should be justified.

Question 1: Write your name and student number.

Solution:

Ryan Lo (101117765)

Question 2: Let $K \geq 3$ be an integer. A K -kite is a graph consisting of a clique of size K and a path with K vertices that is connected to one vertex of the clique; thus, the number of vertices is equal to $2K$. In the figure below, the graph with the black edges forms a 5-kite.



The *kite problem* is defined as follows:

$$\text{KITE} = \{(G, K) : \text{graph } G \text{ contains a } K\text{-kite}\}.$$

Prove that the language KITE is in **NP**.

Solution:

To prove that KITE is in NP, we need to show that there is a polynomial-time algorithm that can verify this.

To verify whether or not (G, K) is in KITE we can do these steps:

1. Check if G contains a clique of size K by checking all possible subsets of K vertices in G and seeing whether they form a complete subgraph or not.
2. If G does not contain a clique of size K , (G, K) cannot be KITE.
3. Otherwise, choose a vertex in the clique and check if there is a path of K vertices in G that is connected to that vertex, by exploring all paths of length K that start at the vertex.
4. If there exists a path of K vertices in G that is connected to that vertex, then (G, K) is in KITE, otherwise it is not.

The runtime of this is in polynomial time, therefore, KITE is in NP.

Question 3: The *clique problem* is defined as follows:

$$\text{CLIQUE} = \{(G, K) : \text{graph } G \text{ contains a clique of size } K\}.$$

Prove that $\text{CLIQUE} \leq_P \text{KITE}$, i.e., in polynomial time, CLIQUE can be reduced to KITE.

Solution:

To prove that $\text{CLIQUE} \leq_P \text{KITE}$, we need to show that there is a polynomial-time reduction from Clique to Kite. We need to show that given (G, K) of Clique, we can construct (G', K') of Kite such that (G, K) is in Clique if and only if (G', K') is in Kite.

Construction of G' :

For each vertex v in G , add a new vertex to G' and connect it to the vertex in G' to form a path of length 1. Then create a new vertex and connect it all to the vertices to form a clique.

The resulting new graph should have $2K$ vertices and constructed in polynomial time. G' contains a K -kite if and only if G contains a clique of size K .

Suppose that $(G', 2K)$ is an instance of Kite, then the clique in the K -kite must include the new vertex and at least $K - 1$ vertices from before. Let S be the set of corresponding vertices in G , then S forms a clique of size K in G , thus G contains a clique of size K .

Therefore, the reduction algorithm constructs $(G', 2K)$ of Kite from (G, K) of Clique in polynomial time, and (G, K) is in Clique if and only if $(G', 2K)$ is in Kite, and $\text{CLIQUE} \leq_P \text{KITE}$.

Question 4: The *subset sum problem* is defined as follows:

$$\text{SUBSETSUM} = \{(S, t) : \begin{array}{l} S \text{ is a set of integers, } t \text{ is an integer,} \\ \exists S' \subseteq S \text{ such that } \sum_{x \in S'} x = t \end{array}\}.$$

The *partition problem* is defined as follows:

$$\text{PARTITION} = \{S : \begin{array}{l} S \text{ is a set of integers,} \\ \exists S' \subseteq S \text{ such that } \sum_{x \in S'} x = \sum_{y \in S \setminus S'} y \end{array}\}.$$

- Prove that $\text{SUBSETSUM} \leq_P \text{PARTITION}$, i.e., in polynomial time, SUBSETSUM can be reduced to PARTITION.
- Prove that $\text{PARTITION} \leq_P \text{SUBSETSUM}$, i.e., in polynomial time, PARTITION can be reduced to SUBSETSUM.

Solution:

To prove that $\text{SUBSETSUM} \leq_P \text{PARTITION}$, i.e., in polynomial time, SUBSETSUM can be reduced to PARTITION, we need to construct S' of Partition.

Given (S, t) , we construct S' :

First we compute the sum of all the elements in S and we can call that $\text{sum}(S)$. Next we double every element in S and let that be S' . Now we compute the sum of all elements in S' and label that $\text{sum}(S')$. Now if $t > \text{sum}(S')$ then add $(t - \text{sum}(S'))$ to S' as a new element, otherwise we add $(\text{sum}(S') - t)$ to S' as new element.

Now we suppose that (S, t) is an instance of SubsetSum. Then there exists a subset $S'' \subseteq S$ such that the sum of the elements in S'' is equal to t . Let $S''' = 2x : x \in S''$. Then the sum of the elements in S''' is equal to $2t$. Since S' contains all the elements of S''' and possibly one more element, the sum of the elements in S' is equal to $2t + a$, where a is either 0 or $|t - \text{sum}(S')|$. Therefore, S' is an instance of Partition.

We now suppose that S' is an instance of Partition. Then there exists a subset $S'' \subseteq S'$ such that the sum of the elements in S'' is equal to $\text{sum}(S')/2$ (if $\text{sum}(S')$ is odd, then we

consider the closest integer to $\text{sum}(S')/2$. Let $S'' = \{x/2 : x \in S'\}$. Then the sum of the elements in S'' is equal to $\text{sum}(S)/2$. Since S'' is a subset of S , the sum of the elements in S'' is also equal to t , which means that (S, t) is an instance of SubsetSum.

Therefore, reduction is correct and $\text{SUBSETSUM} \leq_P \text{PARTITION}$.

We will now prove that $\text{PARTITION} \leq_P \text{SUBSETSUM}$, i.e., in polynomial time, PARTITION can be reduced to SUBSETSUM

Given S , we will need to construct (S', t) :

We need to compute the sum of all elements in S and label it $\text{sum}(S)$ again. Then we compute $\text{sum}(S)/2$ and set that equal to t . Lastly, we let S' be S .

First, suppose that S is an instance of Partition. Then there exists a subset $S'' \subseteq S$ such that the sum of the elements in S'' is equal to $\text{sum}(S)/2$. Since t is equal to $\text{sum}(S)/2$, the sum of the elements in S'' is also equal to t . Therefore, (S', t) is an instance of SubsetSum.

Now we suppose that (S', t) is an instance of SubsetSum. Then there exists a subset $S'' \subseteq S'$ such that the sum of the elements in S'' is equal to t . Since $S' = S$, the sum of the elements in S'' is also equal to $\text{sum}(S)/2$. Therefore, S is an instance of Partition.

Therefore, reduction is correct and $\text{PARTITION} \leq_P \text{SUBSETSUM}$.

Question 5: The *clique and independent set problem* is defined as follows:

$$\text{CLIQUEINDEPSET} = \{(G, K) : \begin{array}{l} \text{graph } G \text{ contains a clique of size } K \text{ and} \\ G \text{ contains an independent set of size } K \end{array}\}.$$

Prove that $\text{CLIQUE} \leq_P \text{CLIQUEINDEPSET}$, i.e., in polynomial time, CLIQUE can be reduced to CLIQUEINDEPSET.

Solution:

Given an instance of Clique (G, K) , we need to construct an instance of the CliqueIndepSet (G', K') such that G contains a clique of size K if and only if G' contains a clique of size K' and an independent set of size K' .

Construction: Let G' be the complement of G , i.e., the graph with the same vertex set as G , but with an edge between any two vertices that are not connected by an edge in G . Clearly, a clique of size k in G corresponds to an independent set of size k in G' . Similarly, an independent set of size k in G corresponds to a clique of size k in G' . Therefore, if we set $K' = |V(G)| - K$, then G contains a clique of size K if and only if G' contains a clique of size K' and an independent set of size K' . The construction is done in polynomial time. Therefore, the reduction from Clique to CliqueIndepSet is in polynomial-time, $\text{CLIQUE} \leq_P \text{CLIQUEINDEPSET}$.

Question 6: Let φ be a Boolean formula in the variables x_1, x_2, \dots, x_n . We say that φ is in *conjunctive normal form* (CNF) if it is of the form

$$\varphi = C_1 \wedge C_2 \wedge \dots \wedge C_m,$$

where each C_i , $1 \leq i \leq m$, is of the following form:

$$C_i = l_1^i \vee l_2^i \vee \dots \vee l_{k_i}^i.$$

Each l_j^i is a *literal*, which is either a variable or the negation of a variable.

The *satisfiability problem* is defined as follows:

$$\text{SAT} = \{\varphi : \varphi \text{ is in CNF-form and is satisfiable}\}.$$

Prove that $\text{CLIQUE} \leq_P \text{SAT}$, i.e., in polynomial time, CLIQUE can be reduced to SAT.

Solution:

Given an instance of Clique (G, K) , we need to construct an instance of the satisfiability problem, which is a Boolean formula in CNF-form φ such that G contains a clique of size K if and only if φ is satisfiable.

Let $V = v_1, v_2, \dots, v_n$ be the set of vertices of G . For each vertex v_i , introduce a Boolean variable x_i . We will construct a Boolean formula φ in CNF-form that is satisfiable if and only if G has a clique of size K .

First, add the clauses $(x_i \vee x_j)$ for every pair of vertices i, j that are connected by an edge in G . This ensures that if two vertices are connected, at least one of them must be in any clique.

Next, add the clauses $\neg x_i \vee \neg x_j$ for every pair of vertices i, j that are not connected by an edge in G . This ensures that if two vertices are not connected, both of them cannot be in any clique.

Finally, add the clauses $(x_{i_1} \vee x_{i_2} \vee \dots \vee x_{i_k})$ for every k -clique in G . This ensures that at least k vertices in the clique must be true, which is only possible if the corresponding Boolean variables are set to true.

Construction is done in polynomial-time, therefore, $\text{CLIQUE} \leq_P \text{SAT}$.