# Algorithms (6470) HW05

Alex Darwiche

July 31, 2024

## Answers

### Q1

(a) Verifier: Determine whether all $k$ sets have a total less than or equal to 1.

```
def verifier(subset, k):
    for each subset:
        total = 0
        for each element of subset:
            total = total + element
        if total > 1:
            return False
    return True
```

(b) Runtime Analysis: This verifier runs through k subsets, and then sums across each subset n times where n is the number of elements in each subset. We can find an upper bound here by saying n is equal to the largest size of a subset. The check if total $>$ 1 is constant time. So this runs in $O(k) * (O(n) + O(1)) = O(kn)$. Given k is a postitve integer constant, this verifier runs in polynomial time and this problem is in $NP$.

(c) This verifier goes through each subset, finds the total of that subset, and returns False if a subset's total is ever over 1.

## Q2

(a) Independent-Set Verifier

```
def verifier(G,V.prime,k):
    for size(V.prime) != k:
        return False

    for vert1 in V.prime:
        for vert2 in V.prime:
            if vert1 != vert2:
                if edge(vert1,vert2) in G.E:
                    return False
    return True
```

(b) Runtime Analysis: This verifier first checks the size of V.prime (V'). Checking the size of a set is $O(|V'|)$ in the worst case where you need to iterate through every element. Then the algorithm checks that size against the constant value K. This check is constant time $O(1)$. Next, there is a nested for loop iterating through the vertices in V' and checking whether the edge between 2 vertices is present in the edges of Graph G. This nested for loop will have a runtime of $O(|V'|^2)$. Next, there is a check of whether vert1 and vert2 are the same, this is constant time $O(1)$. Lastly, we are looking to determine if the edge between vert1 and vert2 is in G.E, this search will take $O(|E|)$ to complete.

(c) So, in totality, this algorithm takes: $O(|V'|) + O(1) + O(|V'|^2) * (O(1) + O(|E|)) = O(E * |V'|^2)$. This verifier algorithm is in polynomial time, and thus the Indepedent-Set algorithm is part of $NP$.

## Q3

(a) Reduce the CLIQUE problem to the INDEPENDENT-SET problem and use the solution to problem 2.

(b) The CLIQUE problem implies that a set of vertices V', size $k$, exists within Graph G, such that all vertices within V' are connected to one another.

(c) The complement of Graph G, is Graph G'. The complement of a graph includes that same vertices, but has all the edges that are not present in G, and none of the edges that are present in G.

(d) For this G', all the vertices of V' are NOT connected to one another, given they were in G, and thus form an independent set.

(e) This shows that we can reduce the CLIQUE problem to the INDEPEDENT-SET problem. Since we know that the CLIQUE problem is NPC, we can also say that the INDEPENDENT-SET problem is NPC.

(f) The transformation from Graph G to Graph G' would require looping through each vertex and creating an edge if one doesn't exists, or removing it if it does exist.

(g) This transformation is polynomial time, so we can conclude that INDEPENDENT-SET $\leq_p$ CLIQUE.

## Q1 (Graduate students only)

(a) First we can see that SUBGRAPH-ISOMORPHISM is part of NP as we can find a verifier that is polynomial time to determine if G is isomorphic to G2. We would simply need to loop through the vertex/edge combinations in G, and confirm they are present in H as well.

(b) We can reduce the CLIQUE problem for this SUBGRAPH-ISOMORPHISM as well.

(c) The CLIQUE problem implies that a set of vertices V', size $k$, exists within Graph G, such that all vertices within V' are connected to one another.

(d) The CLIQUE problem, can be reduced to a Graph G, and a Graph H, where H is the complete graph of V' vertices, representing the clique, size $k$.

(e) So, we are left with 2 graphs and trying to determine if Graph H is an isomorphic subgraph of G.

(f) This now imitates the SUBGRAPH-ISOMORPHISM problem. So to solve SUBGRAPH-ISOMORPHISM in this case, you would need to get the decision from the CLIQUE problem.

(g) Since we know that CLIQUE is NP-Complete, we can now conclude that SUBGRAPH-ISOMORPHISM is also NP Complete. SUBGRAPH-ISOMORPHISM $\leq_p$ CLIQUE.

(h) The transformation from G to H is $O(k^2)$ given that you will need to draw a line from each vertex to another in the set. This is a polynomial time transformation.