CS 325 – Winter 2023
Analysis of Algorithms
Homework 6

JuHyun Kim

Q1. (Search-to-decision for k-SAT)

Input: A k-SAT formula $\phi(x_1, \ldots, x_n)$ on n variables that are satisfiable.

Output: A Boolean assignment $a_1, \ldots, a_n \in \{0, 1\}$ that satisfies ϕ.

```
kSAT( ϕ):
if D(ϕ)==False
        return None
For i←1 to n
        If x_i=1
                ϕ← ϕ_{x_i=1}
                A[i]←True
        else
                ϕ← ϕ_{x_i=0}
                A[i]←False
return A[1..n]
```

If ϕ is unsatisfiable, return None. D(ϕ) is a function that returns True if ϕ is satisfiable and False otherwise. Initialize an array A[1..n] of Boolean values to False.

For each variable x_i in ϕ:

a. If ϕ ∧ xi=1 is satisfiable, set xi to 1 and set A[i] to True.

b. Otherwise, set xi to 0 and set A[i] to False.

Return the assignment A[1..n].

This algorithm works by iteratively assigning values to variables in ϕ that satisfy the formula. It starts by checking if ϕ is satisfiable, and if not, it returns None. If ϕ is satisfiable, it proceeds to iterate over each variable in ϕ and assigns a value to each variable that satisfies ϕ. The algorithm keeps track of assignments A, which is returned at the end.

Q2. (Guards and Streets)

Input: set of locations L={l_1, ... , l_n}, set of streets S = {s_1, ... , s_n}, and number $k \geq 0$

Each s has exactly two distinct location l_i and l_j as its endpoints.

A street $s \in S$: guarded
S : totally guarded
Output:

Yes- It is possible to totally guard S suing at most k (i.e., $\leq k$)guards.
No-To totally guard S, it is necessary to use strictly more than k (i.e., $> k$)guards.

a. Prove that the GuardsAndStreets Problem is in NP.

To show that this problem is in NP, give a polynomial-time verifier that can verify a solution.. The verifier takes two inputs: the original input to the problem, and a certificate (a potential solution to the problem). The verifier should run in polynomial time in the size of the inputs. For each point on the street, check if there exists a guard within the range of vision (within a certain distance) of that point. If there is no guard within range of vision for any point, return "NO". If all points on the street are within the range of vision of at least one guard, return "YES". This verifier runs in polynomial time in the size of the inputs, since it performs a constant-time operation (checking whether a point is within range of vision of a guard) for each point on the street, which is a polynomial number of operations in the size of the inputs. Therefore, we have shown that the GuardsAndStreets problem is in NP. The certificate in this case is simply the set of guard positions, and the verifier checks whether this set covers all points on the street.

b. Give a polynomial-time reduction R from the IndependentSET Problem that we saw in class to the GuardsAndStreets Problem. Conclude that the GuardsAndStreets Problem is NP-hard, and by part (a), also NP-Complete.

Reduction from IndependentSET to GuardsAndStreets.
Input: A graph G = (V,E) and an integer k.
Output: A description of a street with n points and a set of m points where guards are placed.
Construct a new graph G' = (V', E')
Construct a street with n = |V'| vertices, where each vertex corresponds to a vertex in V'.

For each vertex v in G, place a guard at the midpoint of the line segment between v and its corresponding vertex in V. Return the street and the set of all guards as the output.

Assumed that G has an independent set of size k. Then, choose the same k vertices in V and no two of them are neighboring in G. Since each vertex in G is connected to all vertices in V', also no two vertices in the independent set are neighboring in G'. Since each vertex in G is associated with a guard at its midpoint, each vertex in the independent set is within range of vision of at least one guard. So the guard placement satisfies the GuardsAndStreets problem for this street.
Conversely, suppose that there exists a guard placement that satisfies the GuardsAndStreets problem for the street constructed by the reduction. Then, choose the set of vertices that correspond to the guards as an independent set of size k in G. Since each vertex in G is connected to all vertices in V', no two vertices in the independent set are adjacent in G'. Since no two guards are placed on adjacent vertices in the street, each vertex in the independent set is within range of vision of at least one guard. Therefore, the guard placement satisfies the IndependentSET problem for G.

Since the reduction runs in polynomial time and is correct, the GuardsAndStreets problem is NP-hard. Furthermore, since we have also shown that the GuardsAndStreets problem is in NP, it follows that the GuardsAndStreets problem is NP-complete.