Assignment 8

34.2-6 A hamiltonian path in a graph is a simple path that visits every vertex exactly
once. Show that the language HAM-PATH= ({G,u,v}: there is a hamiltonian path from u to v in graph G belongs to NP).

Our aim is to show that the language HAM-PATH can be verified in polynomial time. Input

x be ⟨G, u, v⟩ and certificate y be a sequence of vertices $\{v1, v2, \cdots, vn\}$.

An algorithm A(x, y) verifies HAM-PATH by executing steps:

(a) if $|G.V| = n$;
(b) if $v1 = u$ and $vn = v$;

(c) if $\forall i \in \{1,2,\cdots,n\}, vi \in G.V$;

(d) if $\forall i,j \in \{1,2,\cdots,n\}$, vi != vj;

(e) if $\forall i \in \{1,2,\cdots,n-1\}, (vi,vi+1) \in G.E$;

If any of the above steps fails, return false. Else return True;
Steps (a) and (b) take O(1) time; step (c) takes O(V) time; step (d) runs in O(V 2) time and step (e) runs in O(E) time. Therefore the verification algorithm runs in O(V 2) time.
Hence HAM-PATH $\in$ NP.

34.2-7

Show that the hamiltonian-path problem from Exercise 34.2-6 can be solved in polynomial time on directed acyclic graphs. Give an efficient algorithm for the Problem.

The Hamiltonian Path problem can be solved in polynomial time on directed acyclic graphs (DAGs) because DAGs have a particular structure that simplifies the search for Hamiltonian paths. In a DAG, there are no cycles, which means that vertices can be linearly ordered such that there is a directed edge from vertex i to vertex i+1 for all i in the order. This linear structure makes finding Hamiltonian paths in DAGs relatively straightforward.

Perform a topological sorting of the vertices in the DAG. A topological sort linearly orders the vertices in such a way that if there is a directed edge from vertex u to vertex v, then u comes before v in the order.

Check for Hamiltonian Path: Once you have a topological order of the vertices, check if there is a path from the first vertex to the last vertex in the order. If there is a path, you have found a Hamiltonian path in the DAG. If there is no such path, there is no Hamiltonian path in the DAG.

This algorithm works because the topological sorting guarantees that you visit vertices in an order that respects the directed edges of the DAG, and therefore, a path from the first to the last vertex in the topological order corresponds to a Hamiltonian path.

The time complexity of this algorithm is determined by the topological sorting step, which can be done in linear time, O(|V| + |E|), where |V| is the number of vertices and |E| is the number of edges in the graph. The second step of checking for a path has a linear time complexity. Therefore, the overall time complexity of finding a Hamiltonian path in a DAG is O(|V| + |E|), making it a polynomial-time algorithm for DAGs.

34.4-6

Suppose that someone gives you a polynomial-time algorithm to decide formula satisfiability. Describe how to use this algorithm to find satisfying assignments in polynomial time.

Suppose the algorithm is A, the formula is **Φ** and the variables are(x_1……x_n) .
If A rejects **Φ**, then there is no solution.
Else if A accepts **Φ**, then for each x_i, replace x_i by 0, if the transformed formula **Φ'** is accepted by A, then $x_i=0$, otherwise $x_i=1$.