1) 20 pts
Mark the following statements as **TRUE** or **FALSE**. No need to provide any justification.

[TRUE/FALSE]
Every problem in P can be reduced to 3-SAT in polynomial time.

[TRUE/FALSE]
If there is a polynomial-time algorithm for 2-SAT, then every problem in NP has a polynomial-time algorithm.

[TRUE/FALSE]
If all edge weights are 1, 2, or 3, the shortest path problem can be solved in linear time.

[TRUE/FALSE]
Suppose G is a graph with n vertices and $n^{1.5}$ edges, represented in adjacency list representation. Then depth-first search in G runs in $O(n^{1.5})$ time.

[TRUE/FALSE]
The weight of a minimum spanning tree in a positively weighted undirected graph is always less than the total weight of a minimum spanning path (Hamiltonian Path with lowest weight) of the graph.

[TRUE/FALSE]
If A is in NP, and B is NP-complete, and A $\leq$p B then A is NP-complete.

[TRUE/FALSE]
Given a problem B, if there exists an NP-complete problem that can be reduced to B in polynomial time, then B is NP-complete.

[TRUE/FALSE]
If an undirected connected graph has the property that between any two nodes u and v, there is exactly one path between u and v, then that graph is a tree.

[TRUE/FALSE]
Suppose that a divide and conquer algorithm reduces an instance of size n to four instances of size $n/5$ and spends $\Theta(n)$ time in the divide and combine steps. The algorithm runs in $\Theta(n)$ time.

[TRUE/FALSE]
An integer N is given in binary. An algorithm that runs in time $O(sqrt(N))$ to find the largest prime factor of N is considered to be a polynomial-time algorithm.
Ex: Prime factorization of 84: 2 x 2 x 3 x 7, so the largest prime factor of 84 is 7