Homework 5.
Due: Thursday, July 15, 2021 before 11:59PM via Gradescope. Late submission with no penalty by Friday, July 16, 2021 before 11:59AM.

Problem 1 (SAT variant).

Solution:
Given an input of 3-CNF that each literals appear no more than 3 times we check that the problem is in NP checking if any one of the 3 literals in n claues equal to 1. This takes runtime $O(3n)$ and this is polynomial time. So the problem is in class np.

We now reduce 3-SAT to our problem. We first map a 3-SAT instance to the instance of our problem.
For each variable in the 3-SAT instance that appears in more than 3 claues, let the variable be x, we replace its kth appearance by x_k.
For example, variable x with 4 appearances in 4 clauses will be replaced by x_1, x_2, x_3, x_4.
Then we add clauses:

$$(\bar{x}_1 \vee x_2) \wedge (\bar{x}_2 \vee x_3)...(\bar{x}_k \vee x_1)$$

This is to make sure that each sub variable we create has the same value. As the added claues show, if x is originally true, by making x_1 true, x_2 has to be true according to the second clause, and the rest of the variables are forced to be true. vice versa.
The adding clause part has runtime O(m) because there are at most m clauses and thus at most m variables.

We repeat this for each variable that appear in more than 3 clauses.
This mapping has runtime O(nm) which is in polynomial time, n is the number of variables.

The 3-SAT problem has a satisfying solution if and only if the reduced problem has a solution
Given a truth assignment of the 3-SAT problem, simply setting the created variables x_i to x in our problem gives us a truth solution.
Given a truth assignment of our problem, we can just put those variables back together, this takes O(m) which is polynomial time.

This concludes the reduction.

Problem 2 (Max diameter spanning tree).

Solution:
We reduce from Rudrata path to the MDST problem. We map the instance graph G and pass it to MDST. This has runtime O(n+m) and is polynomial.
We also pass in g = |V|-1 which V is the number of vertices. The reason to pass in |V|-1 is that rudrata path requires the path to go through all vertices while visiting each vertex only once, and that forces the path to go through |V|-1 edges which make the length of the path.

We show that there is a solution for Rudrata path if and only if there is one for the MDST.
Note that the length of the Rudrata path has to be |V|-1, so if a rudrata path exist, the rudrata path will be the diameter making the diameter's length |V|-1.
Similarily, if MDST(G, |V|-1) has a solution, the diameter has length |V|-1 and is a path.

This concludes the reduction.