

CS3230: Assignment for Week 11 Solutions

Due: Sunday, 17th Apr 2022, 11:59 pm SGT.

1. (a) Yes. This would be true if $P = NP$.
(b) Yes. This would be true if $P \neq NP$.
(c) No. If one NP-complete problem can be solved in polynomial time, then all of them can be solved in polynomial time, because they can be reduced to each other in polynomial time.

2. First, the problem 2022-LABEL belongs to NP: a certificate for a YES-instance is a labeling $1, 2, \dots, 2022$ of the graph such that no two adjacent vertices receive the same label. A verifier can check in polynomial time that each edge connects two vertices with different labels.

Next, to show that 2022-LABEL is NP-hard, we reduce from 2021-LABEL. Given an instance G of 2021-LABEL, as in the hint, we add an extra vertex v that is connected to all existing vertices to obtain an instance G' of 2022-LABEL. Clearly, this reduction can be done in polynomial time.

If G is a YES-instance of 2021-LABEL, then by labelling v with 2022, we get a valid labeling of G' with $1, 2, \dots, 2022$. For the opposite direction, suppose G' is a YES-instance of 2022-LABEL. Assume without loss of generality that v is labelled 2022. Since v is connected to all remaining vertices, no other vertex can be labelled 2022. Hence, all other vertices are labelled $1, 2, \dots, 2021$, which gives a valid labeling of G .

3. First, the problem 3-SATTWICE belongs to NP: a certificate for a YES-instance consists of two satisfying assignments, which the verifier can check in polynomial time.

Next, to show that 3-SATTWICE is NP-hard, we reduce from 3-SAT. Given an instance ϕ of 3-SAT, we add a clause $(x \vee x \vee \bar{x})$, where x is a new variable, to obtain an instance ϕ' of 3-SATTWICE. Clearly, this reduction can be done in polynomial time.

If ϕ is a YES-instance of 3-SAT, then by setting x to either True or False and using the same satisfying assignment of ϕ for the remaining variables, we get two satisfying assignments for ϕ' . For the opposite direction, a satisfying assignment for ϕ' immediately yields one for ϕ by ignoring x .