CS3230: Assignment for Week 10

Due: Sunday, 10th Apr 2022, 11:59 pm SGT.

Please upload PDFs containing your solutions (hand-written & scanned, or typed) by 10th Apr, 11:59 pm to `Assignments/Assignment10/Submissions`. Name the file **Assignment10_SID.pdf**, where SID should be replaced by your student ID.

You may discuss the problems with your classmates or read material online, but you should write up your solutions on your own. Please note the names of your collaborators or online sources in your submission; failure to do so would be considered plagiarism.

1. (1 point) Suppose someone gives you a polynomial-time algorithm to decide satisfiability of 3-SAT formulas. Describe how to use this algorithm to find a satisfying assignment for a given 3-SAT formula in polynomial time (when such an assignment exists).

2. (1 point) In the problem LONGEST PATH, you are given an undirected graph and two vertices, and your goal is to return the number of edges in a longest simple path between the two vertices. In the decision version, LONGEST PATH LENGTH, you are additionally given a nonnegative integer k, and your goal is to decide whether there exists a simple path of length at least k between the two vertices.

   Prove that there is a polynomial-time algorithm for LONGEST PATH if and only if there is one for LONGEST PATH LENGTH.

   (Note: In a *simple path*, all vertices are distinct; this is often referred to as just a *path*.)

3. (1 point) In the problem PARTITION, you are given a list of (not necessarily distinct) positive integers $a_1, \ldots, a_n$, and the goal is to determine whether you can partition them into two disjoint parts with the same sum.

   (a) Design and analyze an algorithm running in time $O(nM)$ for this problem, where M is the sum of all numbers. (You may either do this directly, or use the relation to the problem KNAPSACK covered in Week 12 tutorial.)

   (b) Is the algorithm in part (a) a polynomial-time algorithm for PARTITION?