**Carnegie Mellon University** Instructor: Venkatesan Guruswami
15-252: More GITCS Term: Spring 2020

# Problem Set 9

Due date: Saturday, March 28

## Instructions

- You are allowed to collaborate with one other student taking the class, or do it solo.
- Collaboration is defined as discussion of the lecture material and solution approaches to the problems. Please note that *you are not allowed to share any written material and you must write up solutions on your own.* You must clearly acknowledge your collaborator in the write-up of your solutions.
- Solutions must be typeset in LaTeX and emailed to the TA.
- You should not search for solutions on the web. More generally, you should try and solve the problems without consulting any reference material other than what we cover in class and any provided notes.
- Please start working on the problem set early. Though it is short, the problem(s) might take some time to solve.

1. (a) For a (simple, undirected) graph $G = (V, E)$, define its "square" $G^2$ as follows. The vertices of $G^2$ consist of ordered pairs of vertices of $G$, i.e., the vertex set is $V \times V$. Two pairs $(u_1, u_2)$ and $(v_1, v_2)$ are adjacent in $G^2$ if any of the following hold:

   i. Both $(u_1, v_1) \in E$ and $(u_2, v_2) \in E$.
   ii. $u_1 = v_1$, $(u_2, v_2) \in E$.
   iii. $u_2 = v_2$, $(u_1, v_1) \in E$.

   Is the following statement true or false: *For every graph $G$, the size of the largest clique in $G^2$ is equal to the square of the size of the largest clique is $G$.* Prove your answer.

   (b) Suppose that there is a polynomial time algorithm $A$ that on any input graph $G$, finds a clique of size at least 1% of the largest clique in $G$. Show how can one use $A$ as a subroutine and design a polynomial time algorithm $B$ that finds a clique of size at least 99% of the largest clique in any input graph.

   Hint: Use the previous part.

2. In this exercise, you will see a general form of the reduction from 3-SAT to CLIQUE that works with any constraint satisfaction problem in the place of 3-SAT.

   Let $P : \{0,1\}^k \to \{0,1\}$ be a predicate and CSP($P$) be the associate constraint satisfaction problem. An instance $\mathcal{I}$ of CSP($P$) consists of a set of variables $V$ and a collection $\mathcal{C}$ of $m$ constraints (for some positive integer $m$ and indexed by $j \in \{1, 2, \ldots, m\}$) of the form $P(\tau_1^{(j)}, \tau_2^{(j)}, \ldots, \tau_k^{(j)})$ for some tuple $\tau^{(j)} \in V^k$ of $k$ variables from $V$. For any assignment $\sigma : V \to \{0,1\}$, we can count the number of constraints, call it $N(\mathcal{I}, \sigma)$, of $\mathcal{I}$ that are satisfied by the values assigned by $\sigma$ to its variables. Let $\mathrm{OPT}(\mathcal{I})$ be the maximum over all assignments $\sigma : V \to \{0,1\}$ of $N(\mathcal{I}, \sigma)$.

We now map an instance $\mathcal{I}$ of CSP($P$) to a graph $H = (W, E)$ as follows. Suppose $\mathcal{I}$ has $m$ constraints. The vertex set $W$ will consist of $m$ disjoint parts $W_1, W_2, \ldots, W_m$, one corresponding to each of the $m$ constraints of $\mathcal{I}$.

The vertices in $W_j$, $1 \le j \le m$, will correspond to assignments to the $k$-tuple $\tau^{(j)}$ of variables that *satisfy* the $j$'th constraint of $\mathcal{I}$. (So all $W_j$'s will have equal size, equal to $|P^{-1}(1)|$, the number of assignments in $\{0,1\}^k$ that satisfy $P$.)

There will be no edges amongst vertices in the same $W_j$, i.e., each $W_j$ is an independent set. A vertex $a \in W_j$ and $b \in W_{j'}$ for two parts $j \neq j'$ are adjacent in $H$ if the assignments corresponding to $a$ and $b$ are *consistent* on the variables that belong to both the tuples $\tau^{(j)}$ and $\tau^{(j')}$. (In particular, if the tuples $\tau^{(j)}$ and $\tau^{(j')}$ are disjoint, then all edges between $W_j$ and $W_{j'}$ are present in $H$.)

Prove that the size of the largest clique in $H$ equals OPT($\mathcal{I}$).