CS 580-Spring 2019
Instructor: Jeremiah Blocki
TAs: Hamidreza Khorasgani, Akash Kumar
I collaborated with (...). I affirm that I wrote the solutions in my own words and that I understand the solutions I am submitting.

Homework 1

Due Date: January 24, 2019 at 11:59PM on Gradescope.

Question 1

1. Show the following using induction. Consider the following graph $H_n = (V_n, E_n)$. Its vertex set is $V_n = \{0,1\}^n$ – that is every bitstring of length n is a vertex. A pair of vertices $u, v \in \{0,1\}^n$ is connected by an edge (i.e., $\{u, v\} \in E_n$) if u and v differ in exactly one bit. For example, if $n = 3$ then $u = (0,0,0)$ and $v = (1,0,0)$ would be connected by an edge, but v and $w = (1,1,1)$ were not connected by an edge. Use the principle of mathematical induction to show that the number of edges in the graph is $|E_n| = n2^{n-1}$.

2. n friends get together at a bar. It takes exactly 3 friends to participate in a cheer. Various (not necessarily all) triples of friends engage in cheers. For an individual x we use $\texttt{CheerCount}(x)$ to denote the number of times individual x participates in a cheer. For $i \in \{0,1,2\}$, let $S_i = \{x \, : \, i = \texttt{CheerCount}(x) \mod 3\}$ denote the set of people x that have cheered i times or $i+3$ times or $i+6$ times etc... For example, if individual x participated in 8 cheers then $x \in S_2$ and if individual y participated in 12 cheers then $y \in S_0$. Show that the number $2|S_2| + |S_1|$ is a multiple of 3.

Question 2

Consider an instance I of the stable marriage problem with n men and n women. For a person (man or woman) x with preference list P_x define $P_x(1/2)$ to be the first $n/2$ persons (women/men) in the list P_x (to simplify notation we will assume that n is even). Suppose that we use Gale Shapley algorithm to find a stable matching. If a person (man or woman) x is paired off with another person $y \in P_x(1/2)$, the Gale Shapley matching is said to be *good for x*. Show that there is at least one person x such that Gale-Shapley matching is *good for x*. You may not make any assumptions about the preference lists.

Question 3

In class we introduced big-O notation $f(n) \in O(g(n))$, $f(n) \in \Omega(g(n))$ and $f(n) \in \Theta(g(n))$. In computer science it is also common to use "little-o" notation. In particular, we say that $f(n) \in o(g(n))$ if for all positive constants $c > 0$ there exists a number N such that for all $n \geq N$ it holds that $f(n) < c \times g(n)$. Similarly, we say that $f(n) \in \omega(g(n))$ if for all positive constants $c > 0$ there exists a number N such that for all $n \geq N$ it holds that $f(n) \geq c \times g(n)$.

Formally prove or disprove each of the following claims. You may assume that for all $n \geq 2^{2^{2019}}$ we have $f(n) \geq 1$ and $g(n) \geq 1$.

1. $h(n) \in \Theta\left(\max\{f(n)^2, g(n)^2\}\right)$ where $h(n) = f(n) \times g(n)$
2. $f(n) \in o\left(g(n)\right)$ if and only if $g(n) \in \omega\left(f(n)\right)$.
3. If $h(n) \in \Theta\left(\log n\right)$ where $h(n) = \ln f(n)$ then $f(n) \in O(n)$.
4. Suppose that $f(n) \in \omega(1)$ then $f(n) \in o\left(f(n)^2\right)$.
5. If $f(n) \notin O\left(g(n)\right)$ then $g(n) \in O\left(f(n)\right)$.
6. $f(n) \in o\left(g(n)\right)$ implies that $h(n) \in O\big(2^{g(n)}\big)$ where $h(n) = 2^{f(n)}$.
7. If $f(n) \in \Theta(h(n))$ and $g(n) \in o\left(h(n)\right)$ then $f(n) \in \omega\left(g(n)\right)$.

Question 4

Suppose that an n-node undirected graph $G = (V, E)$ contains two nodes s and t such that the distance between s and t is strictly greater than $n/2$. Show that there exists some node $v \neq s, t$ such that deleting v from G destroys *all* paths from s to t. Give an algorithm running in time $O\left(n + m\right)$ to find such a node.