Princeton Univ Spring '26	COS 423: Theory of Algorithms
	Homework 2
Out: *Feb 16*	

Problems:

§0 The following questions are more of a "concept check". You don't have to formally write down your answers (but you are welcome to).

(a) If you are to run maximum matching algorithms (augmenting path or blossom) on weighted graphs, would you get a matching with maximum "total" weights?

(b) If you are to add one edge to the graph, how might the MST change? What if you are to delete one edge?

(c) If we have a promise that the shortest paths from s to any other vertex t use at most $d = O(1)$ edges, how fast can you compute shortest paths from s?

§1 The component graph (referred to as G' in Lecture 6) is the graph formed by SCCs being the nodes and the inter-SCC edges being the edges.

(a) Given a directed graph G, design an algorithm to compute its component graph G'. Make sure that G' has no multi-edges (more than one edge between the same pair of vertices). Your algorithm should run in $O(m + n)$ time.

(b) Given a directed graph $G = (V, E)$, design an algorithm to compute a graph $H = (V, E_H)$ on the same vertex set V such that H has the same strongly connected components and the component graph as G, and $|E_H|$ is minimized. Your algorithm should run in $O(m + n)$ time. Note that H does not need to be a subgraph of G.

§2 Recall one phase of Hopcroft-Karp for bipartite maximum matching on $G = (X \cup Y, E)$: Run BFS from all free vertices in X together, and obtain $d[v]$ for each vertex v, the minimum length of alternating paths from a free vertex to v; Let k be the length of the shortest augmenting path; Find a maximal set of *vertex disjoint* augmenting paths of length k and augment along all of them together.

(a) Design an algorithm that executes one phase of Hopcroft-Karp in $O(m)$ time, and prove that it is correct.

(b) Prove that the length of the shortest augmenting path k must increase after one phase.

§3 Consider a bipartite graph $G = (X \cup Y, E)$. Suppose $|X| = |Y| = n$, a perfect matching is a matching of size n (note the graph has $2n$ vertices).

(a) (*Hall's theorem.*) Suppose G satisfies *Hall's condition*: for every $A \subseteq X$, $|A| \leq |N(A)|$, where $N(A) \subseteq Y$ is the set of neighbors of A, $N(A) = \{y \in Y : \exists x \in A, (x, y) \in E\}$. Prove that any matching of size less than n has an augmenting path. Hence, G must have a perfect matching.

(b) Suppose for some $d \geq 1$, every vertex in G has degree *exactly* d. Prove that the edge set E can be decomposed into d disjoint perfect matchings.

§4 (optional) Given a *directed acyclic graph* G, a path-cover is a set of paths in G such that every vertex of G is included in *exactly* one path. Paths may be of any length, including 0. Design a polynomial time algorithm that computes a path-cover with the fewest possible paths.