1. Let $G$ be a directed graph with two special vertices $s$ and $t$. Any two directed paths from $s$ to $t$ are called vertex-disjoint if they do not share any vertices other than $s$ and $t$. Prove that the maximum number of directed vertex-disjoint paths from $s$ to $t$ is equal to the minimum number of vertices whose removal ensures that there are no directed paths from $s$ to $t$.

2. Let $G$ be a directed graph with capacities on its edges and two special vertices $s$ and $t$. The capacity of a directed path from $s$ to $t$ is the smallest of the capacities of edges on the path. Give an efficient algorithm to find a path from $s$ to $t$ of maximum possible capacity.

3. We say that a cut is within $k$ times the mincut if the number of edges in the cut is within $k$ times the number of edges in a mincut. Suppose that $k$ is half an integer, i.e. $2k$ is an integer. Then show that in any undirected graph, the number of cuts within $k$ times the mincut is fewer than $n^{2k}$.

4. Let $P$ range over the set of $s-t$ paths for two vertices $s,t$ of a given graph. Let $C$ range over cuts that separate $s$ and $t$. Then show that

$$\max_P \min_{e \in P} c_e = \min_C \max_{e \in C} c_e.$$  

Here $c_e$ is the capacity of edge $e$.

5. Suppose that the maximum flow algorithm at each step augments on an augmenting path that has the least number of reverse arcs, i.e., flow will be sent backwards on these arcs in the augmenting path. Give a bound on the maximum number of augmentations performed by the algorithm.