1. The Max-Flow Min-Cut theorem is heavy discussed in CS 122A,B and CS222 and so will not be proved in CS225 (but see the notes online for definitions of flow and edge cuts, and a proof, through the Ford-Fulkerson algorithm, of the MFMC theorem for rational capacities).

One of the consequences of the MFMC theorem is that in a directed graph, the maximum number of edge-disjoint directed paths from a node $s$ to a node $t$ is equal to the minimum number of edges needed to remove to disrupt every directed $s$ to $t$ path.

Use this version of the Max-Flow Min-Cut theorem to prove that in a directed graph, the maximum number of node-disjoint paths from a node $s$ to a node $t$ (the paths have $s$ and $t$ in common, but no other nodes) equals the minimum number of nodes (excluding $s$ and $t$) needed to remove to disrupt every directed $s$ to $t$ path.

This is a very easy problem, but introduces the more difficult undirected version in problem 3.

Answer: Let $G$ denote the original graph. Convert any node $v$ of $G$ into two nodes $v_1$ and $v_2$ and put a directed edge from $v_1$ to $v_2$ with capacity equal to $w(v)$. Next, for every edge that was directed into $v$, now direct it into $v_1$, and for every edge that was directed out of $v$, now direct it out of $v_2$. The capacities of all original edges are infinite. Let $g'$ denote the new graph. Then a maximum flow s-t flow in $G'$ specifies a maximum s-t flow in $G$. Further, since all original edges in $G$ have infinite capacity in $G'$, the minimum s-t cut in $G'$ consists only of new edges. Therefore the minimum cut in $G'$ can be considered as a set of nodes in $G$, with total capacity exactly the same as the minimum edge capacity in $G'$. Therefore, by the Max-Flow-Min-Cut theorem (stated for edge capacities) it follows that the maximum s-t flow equals the minimum total weight of a set of nodes whose removal disrupts all s-t paths in $G$.

2. Let $G$ be an undirected, connected graph and let $a$, $b$ be two nodes that are not adjacent in $G$. An $(a,b)$ separator in $G$ is a subset $S$ of nodes whose removal from $G$ disconnects $a$ and $b$. That is, If $G(S)$ is the graph resulting from removing $S$ from $G$, then $a$ and $b$ are in different connected components of $G(S)$.

A minimal $(a,b)$ separator $S$ is an $(a,b)$ separator such that no proper subset of $S$ is also a $(a,b)$ separator. Note that there can be minimal $(a,b)$ separators of different sizes, so ‘minimal’ is not the same as ‘minimum size’.
2a. Claim: Let $S$ be a $(a, b)$ separator and let $C_a$ and $C_b$ be the connected components of $G(S)$ that contain $a$ and $b$ respectively. Then $S$ is a minimal $(a, b)$ separator if and only if every node in $S$ is adjacent to some node in $C_a$ and to some node in $C_b$.

Prove, or explain, this claim.

Answer: First, there is no path from a node $v$ in $S$ to a node in $C_a$, consisting of an edge to a node in $C_b$ and then a path to $C_a$ without going back to $S$, because there are no edges between $C_b$ and $C_a$. So, if there is a node $v$ in $S$ that is not adjacent to some node in $C_a$, then every path between $v$ and any node in $C_a$ must go through some other node in $S$. Therefore any path from $a$ to $b$ that goes through $v$ must contain a node in $S - v$, contradicting the minimality of $S$. Hence $v$ must be adjacent to some node in $C_a$, and similarly, adjacent to some node in $C_b$.

Conversely, if a node $v$ in $S$ is adjacent to some node in $C_a$ and some node in $C_b$, then there would be a path from $a$ to $b$ through $v$ after the removal of only $S - v$ (where certainly the nodes in $C_a$ ($C_b$) would still be in the same connected component). Hence every node in $S$ is required in order to separate $a$ from $b$, so $S$ is a minimal $a, b$ separator.

2b. Does the following algorithm find a minimal $(a, b)$ separator $S$ in $G$, assuming there is no $(a, b)$ edge in $G$? Explain.

Let $A(a)$ be the set of nodes adjacent to $a$ in $G$. Remove $A(a)$ from $G$ and let $C_b$ be the connected component of the resulting graph $G(A(a))$ containing node $b$. Then find every node in $A(a)$ that is adjacent to at least one node in $C_b$. The set of such nodes, $S$, is the claimed minimal $(a, b)$ separator.

Answer: Yes it does work. First, $A(a)$ is an $a, b$ separator, since any $a, b$ path must begin with an edge from $a$ to a node in $A(a)$. Every $a, b$ path $P$ must go from $a$ to a node in $A(a)$, must leave $A(a)$ for the last time at some point. At that point $P$ must go directly to a node in $C_b$, since if it goes back to $a$, or to another component of $G(A(a))$, then $P$ must go back to $A(a)$, a contradiction. So, when $P$ leaves $A(a)$ for the last time, it must leave via a node in $S$, and hence $S$ is an $a, b$ separator.

By construction, every node in $S$ is adjacent to some node in $C_b$, and is adjacent to $a$, which is in $C_a$. Hence every node in $S$ is adjacent to a node in $C_b$ and a node in $C_a$, and so by problem 2a, $S$ is a minimal $a, b$ separator.

3. Next we are interested in finding the size of a minimum-size $(a, b)$ separator, call it $N(a, b)$. Menger’s theorem relates $N(a, b)$ to the maximum
number of node-disjoint $a - b$ paths (which share only the endpoints) but does not give an algorithm to find $N(a, b)$.

The posted material titled (Note on vertex connectivity) describes a method for finding $N(a, b)$, and an $(a, b)$ separator of that size, using a single network flow computation.

Study the method then explain how and why it works. Remember that algorithms that compute a maximum $s,t$ flow also compute a minimum $s,t$ edge cut.

Answer posted on the class site.

4. The node connectivity of an undirected, connected graph which is not a complete graph, is the minimum number of nodes needed to remove in order to disconnect the graph.

Prove that if a graph $G$ has node-connectivity $k$ and an extra node $x$ is added to $G$, and $x$ is made adjacent to $k$ nodes of $G$, then the new graph will have node-connectivity at least $k$.

Answer: Let $H$ denote the new graph. Consider any subset $S$ of $k - 1$ nodes in $H$. We want to show that $H - S$ is connected, so the node-connectivity of $H$ is at least $k$.

Assume to the contrary, that the removal of $S$ disconnects $H$. If $y \in S$, then the removal of $S - y$ disconnects $G$. But that contradicts the assumption that $G$ has connectivity $k$.

If $y \notin S$, then, since $y$ has $k$ neighbors, but $|S| = k - 1$, there is a neighbor of $x$, call it $z$, that is not in $S$. Now $G - S$ is connected, since $G$ has connectivity $k$, so there is a path from $y$ to any node $v$ in $H - S$, starting with the edge $yz$ and then following with the path from $z$ to $v$ in $G - S$. Hence $H - S$ is connected, and so $H$ has connectivity at least $k$.

5a. Given the result from problem 4, the Fan Lemma is an immediate result using one variant of Menger’s theorem. The Fan Lemma states that if a graph $G$ has node-connectivity at least $k$, then for any set $S$ of $k$ nodes in $G$, and another node $v \notin S$, there is a set of $k$ node-disjoint paths from $v$ to nodes in $S$, where the paths only share node $v$. Give a two or three line proof of the Fan Lemma.

Answer: Add a new node $x$ to $G$ and connect it to all the nodes in $S$. By problem 4, the new graph has connectivity at least $k$, so by Menger’s theorem, there are at least $k$ node-disjoint paths from $v$ to $x$. But each of
these \( k \) paths must go through a node in \( S \), and since \( |S| = k \), there is a set of \( k \) node-disjoint paths from \( v \) to nodes in \( S \) that share only node \( v \).

5b. Prove the following claim: Let \( S \) be a set of \( k \) nodes in a graph \( G \) with node-connectivity \( k \geq 2 \). Then there is a cycle \( C \) in \( G \) that contains all of the nodes in \( S \). Hence if \( G \) has node-connectivity \( k \), then the largest cycle in \( G \) has at least \( k \) nodes.

Note that the cycle might contain nodes not in \( S \). Induction on \( k \) is a sensible way to try to prove the claim.

Answer: The basis is for \( k = 2 \). Let the set of two nodes be \( \{x, y\} \). By Menger’s theorem there are two node-disjoint paths between \( x \) and \( y \), and together they form a cycle that contains \( x \) and \( y \). So now assume the claim is true for node connectivity up to \( k - 1 \) and consider the case of node-connectivity \( k \). Let \( S \) be a set of \( k \) nodes, and let \( x \in S \). Since \( G \) is \( k \) connected, it is \( k - 1 \) connected, so there is a cycle \( C \) containing the nodes in \( S - x \). If \( C \) only contains those nodes, then it only has \( k - 1 \) nodes, and so by the Fan Lemma, there are node-disjoint paths from \( x \) to the nodes in \( C \). Then take any two adjacent nodes on \( C \), and use the two node-disjoint paths from \( x \) to those nodes to extend \( C \) to a cycle through all the nodes of \( S \).

If \( C \) has more than \( k - 1 \) nodes, then let \( z \) be a node on \( C \) not in \( S - x \). By the Fan Lemma, there are \( k \) node-disjoint paths from \( x \) to the \( k \) nodes of \( S - x + z \). Let \( P \) be the path in that set of \( k \) paths, that goes from \( x \) to \( z \). Since all the nodes in \( C \cap S \) are endpoints of \( k - 1 \) of these paths, path \( P \) cannot go through any of the nodes in \( C \cap S \). However, \( P \) might go through some node of \( C \) not in \( S \). Let \( w \) be the first node on \( C \) that path \( P \) (starting from \( x \)) encounters (\( w \) might be \( z \)). Node \( w \) must be in a subpath on \( C \) between two consecutive nodes, call them \( p \) and \( q \), in \( S \). So we have node disjoint paths from \( x \) to \( w \) and from \( x \) to \( p \). We can therefore create a cycle containing all the nodes of \( S \) by going from \( x \) to \( w \), then to \( q \) and all the other nodes of \( C \cap S \), ending at \( p \), and then back to \( x \).

Question: In the proof of the induction step, it almost feels like we did not need the assumption that \( G \) is \( k \) connected. That is, it almost seems that we could have changed the argument to include \( x \) in the cycle, with only the assumption that \( G \) is \( k - 1 \) connected. That of course, would be wrong. But where does the proof break down? Sounds like a good exam-type question.