1. For any $s$-$t$ cut $(A, B)$, the value of a flow $f$ (from $s$ to $t$) is equal to the amount of flow leaving $A$ minus the amount of flow returning to $A$. Therefore, the value of any flow is at most the capacity of any cut (the total capacity of edges crossing from $A$ to $B$).

2. Ford Fulkerson returns a flow with value equal to the capacity of a cut, and is therefore of maximum possible value (thus also “Max-Flow = Min-Cut”)
   (a) F-F terminates when there is no $s$-$t$ path using edges with positive residual capacity
   (b) If we look at the set of vertices reachable from $s$ using only edges of positive residual capacity $A^*$ when F-F terminates, the edges leaving $A^*$ must be saturated and there must not be any flow returning to $A^*$.
   (c) Therefore, the value of the flow is equal to the capacity of the corresponding cut.
   (d) Moreover, we can therefore find a minimum cut by finding this set $A^*$ using reachability in the residual graph – also in total time $O(C|E|)$ where $C$ is the total capacity of edges leaving $s$.

3. We can solve the Maximum Bipartite Matching problem by reducing it to (Integer) Maximum Flow, which is solved by F-F.
   (a) A bipartite graph is one that has a vertex set consisting of two disjoint sets $L$ and $R$ such that all edges cross between $L$ and $R$. A matching is a set of edges that do not share any endpoints.
   (b) Maximum Bipartite Matching: Given a bipartite graph, find a matching of maximum size.
   (c) We will solve maximum bipartite matching using the following algorithm: add a start vertex $s$ with edges to every vertex in $L$, orient the original edges to point from $L$ to $R$, and add a vertex $t$ with an edge from every vertex in $R$. Give all of these edges capacity 1. Now run F-F to obtain an integer maximum flow in this graph. Return the set of edges crossing from $L$ to $R$ with flow 1.
   (d) This algorithm runs in time $O(|V||E|)$.
   (e) Given any matching of size $k$ in the original graph, there is a flow in this graph of value $k$. Therefore the maximum integer flow has value at least $k$.
   (f) Next time, we will show that the set of edges returned by our algorithm is a matching with size at least the value of the flow. In turn, this is therefore a matching that is at least as large as any other matching, i.e., a matching of maximum size.