Homework may be completed in group of size with at most four students. You’re not allowed to use material from the web or talk about the homework with anybody outside your collaboration group (aside from the lecturer or TA.)

Solutions should be typed and uploaded as a pdf to gradescope.com.

To get full marks, answers must be sufficiently detailed, supported with rigorous proofs (of both correctness and running time analysis). Faster algorithms will typically get more marks than slower algorithms.

**Question 1.** Suppose we have an unsorted list of \(n\) distinct values. Consider the following sorting algorithm: if the list is currently unsorted, pick the first two adjacent entries in the list, \([\ldots, a, b, \ldots]\) such that \(a > b\) and swap the position of \(a\) and \(b\). Repeat until the list is sorted. If the list is originally ordered randomly, what is the expected number of swaps necessary before the list is in sorted order.

**Question 2.** Let \(G\) be an undirected graph \(G = (V, E)\) with \(n\) nodes and \(m\) edges.

1. Let \(S\) be a random subset of the nodes, i.e., each node is independently added to \(S\) with probability \(1/2\). What is the expected value and variance of the size of the cut \((S, V \setminus S)\)?
2. Prove that it is possible to partition the nodes of \(G\) into four groups such that at least \(3m/4\) of the edges are cut, i.e., the endpoints are in different groups.

**Question 3.** Consider a bipartite graph \(G = (L \cup R, E)\) where \(|L| = |R| = n\). For any \(S \subseteq L\), let \(\Gamma(S) = \{v \in R : (u, v) \in E \text{ for some } u \in S\}\). Use the max-flow min-cut theorem to prove that there is a matching of size \(n\) iff \(|\Gamma(S)| \geq |S|\) for all \(S \subseteq L\).

**Question 4.** In this question, we consider an alternative to the Edmonds-Karp heuristic for improving the running time of the Ford-Fulkerson algorithm. Rather than finding the augmenting path with the shortest number of hops, the idea was to find the augmenting path with the largest bottleneck capacity; we will call such a path the **biggest** augmenting path. Note that biggest is not the same as longest.

1. Design an efficient algorithm for finding the biggest augmenting path in a residual network.
2. Let \(f\) be the current flow in a capacitated graph \(G\) and let \(g\) be the optimal flow in the residual network \(G_f\). Prove that \(|f| + |g|\) is the size of the optimal flow in \(G\).
3. Prove the best bound you can on the number of iterations of the Ford-Fulkerson algorithm if you use the biggest augmenting path in each round. You may assume all capacities are integers. **Hint:** You may use the following fact without proof: If \(|g|\) is the optimal flow in residual network \(G_f = (V, E_f)\) then the biggest augmenting path in \(G_f\) has bottleneck capacity at least \(|g|/|E_f|\).

**Question 5.** Suppose all the students in the class put their names into a hat and then each student picks out a random name from the hat. Each student needs to buy a present for the student whose name they pick. Without loss of generality you may assume there are \(n\) students in the class.

1. What’s the expected number of people who have to buy presents for themselves?
2. What’s the expected number of “cycles” in the graph where each student is a node and there is a directed edge from \(v_i\) to \(v_j\) if the \(i\)th student picks the \(j\)th student’s name. Note
that is a student picks her own name, that counts as a cycle. Also, if two students pick each other’s name, that also counts as a cycle.