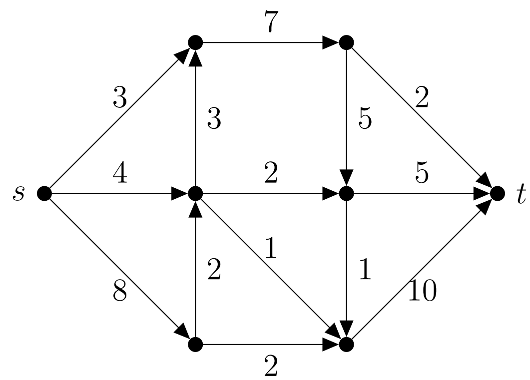
MATH-566 **HW 6**

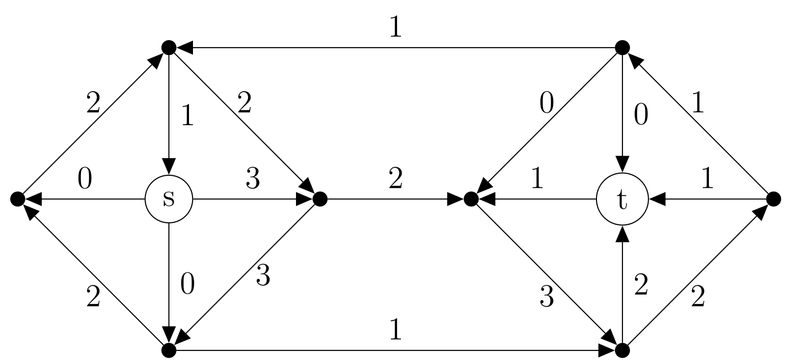
Due **Oct 26** before class. Just bring it before the class and it will be collected there. Solve any 6 out these 7 problems.

1: (*Shortest path and its dual*)
Consider the graph below.

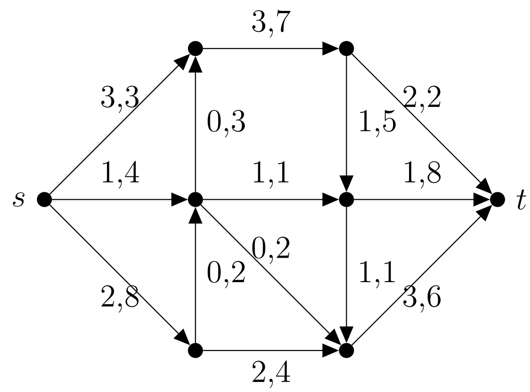


Find a shortest path and prove optimality using duality (find dual LP and its optimal solution).

2: (*Decomposing a flow*)
Consider the network below with given edge values, forming an integer feasible flow. Find a list of path and cycle flows whose sum is this flow.

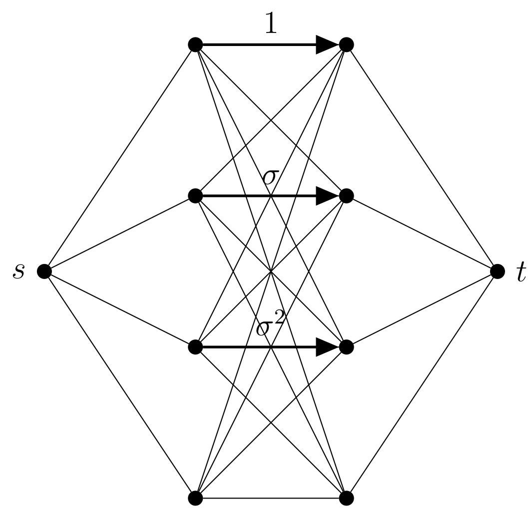


3: (*Augmenting paths*)
Consider the network below with given capacity and flow values. (The edge label f, u means flow-value f and capacity u.) Find augmenting paths and augment the flow to a maximum flow. Provide the list of residual graphs AND augmenting paths. It other words, run Ford-Fulkerson algorithm.



4: (*Combining cuts*)
Let (G, u, s, t) be a network, and let $\delta^+(X)$ and $\delta^+(Y)$ be minimum s-t-cuts in (G, u). Show that $\delta^+(X \cap Y)$ and $\delta^+(X \cup Y)$ are also minimum s-t-cuts in (G, u).

5: (*Ford-Fulkerson algorithm may not finish*)
Show that in case of irrational capacities, the Ford-Fulkerson algorithm may not terminate at all. Hint: See the Korte book (in particular exercises on page 199.). It contains the following network:



Where $\sigma = \frac{\sqrt{5}-1}{2}$. Note that σ satisfies $\sigma^n = \sigma^{n+1} + \sigma^{n+2}$. All other capacities are 1.

6: (*Red-Blue Edges for MST*)
Red-Blue meta algorithm for MST. Let G be a graph and w be a weight assignment to $E(G)$. Assume that all weights are distinct. Start with all edges being uncolored. Apply the following rules as long as possible.

- if $e \in E$ is in a cycle C where e is the heaviest edge, color e red
- if there is a cut where $e \in E$ is the lightest edge, color e blue.

Claim is that blue edges form a minimum spanning tree.

- Show that red edge cannot be in MST.
- Show that blue edge must be in MST.
- Show that blue edges form a tree
- Show that every edge gets colored.
- Show that no edge satisfies both red and blue criteria. (i.e. every edge has one color).

7: (*Edmons-Karp*)
Implement Edmons-Karp algorithm and run it on the network from question three. Print the sequence of augmenting paths used by your implementation. Print the flow and its value.