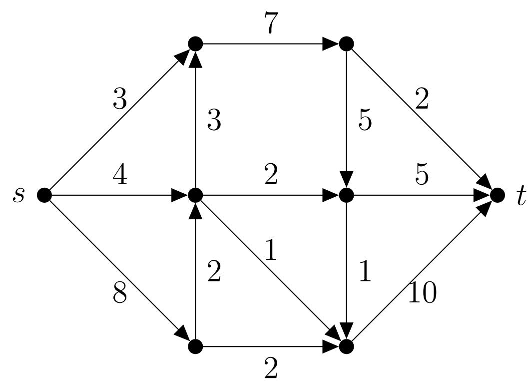
MATH-566 HW 6

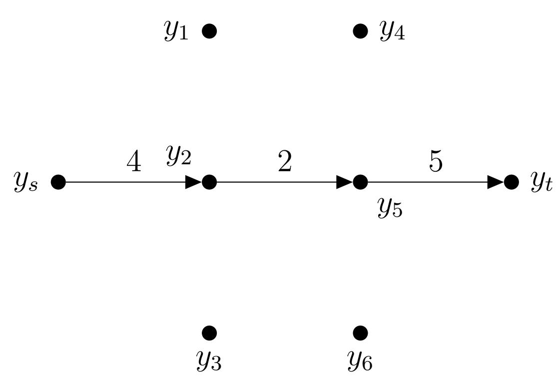
Due **Oct 26** before class. Just bring it before the class and it will be collected there. Solve any 6 out these 7 problems.

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).

Solution:
Here is the depicted solution of the shortest path. The length/cost of the path is 11. We also added labels to vertices in order to write solution to the dual. Denote the cost function on edges by c.



Recall that the dual has form

$$(D)\begin{cases}\text{mazimize} & y_t - y_s\\ \text{subject to} & y_j \le y_i + c(ij) \text{ for all edges } ij\\ & y_s = 0\end{cases}$$

Possible solution is

$$y_s = 0, y_1 = 3, y_2 = 4, y_3 = 8, y_4 = 10, y_5 = 6, y_6 = 5, y_t = 11$$

The value of the solution is 11, which proves the optimality of the shortest path. Or it could be written as

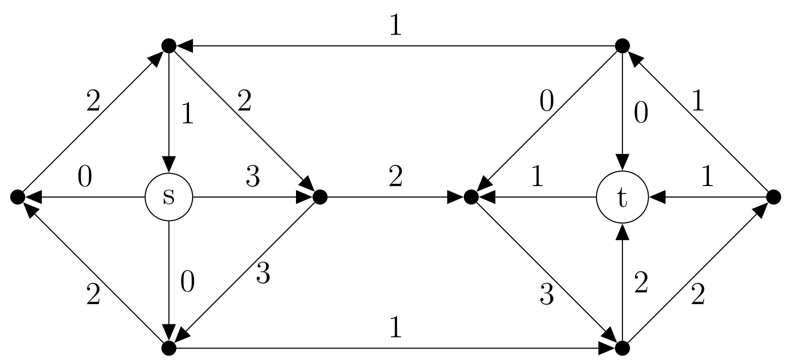
$$(D)\begin{cases} \text{mazimize} & y_s - y_t \\ \text{subject to} & y_i - y_j \leq c(ij) \text{ for all edges } ij \\ & y_s = 0 \end{cases}$$

with solution Possible solution is

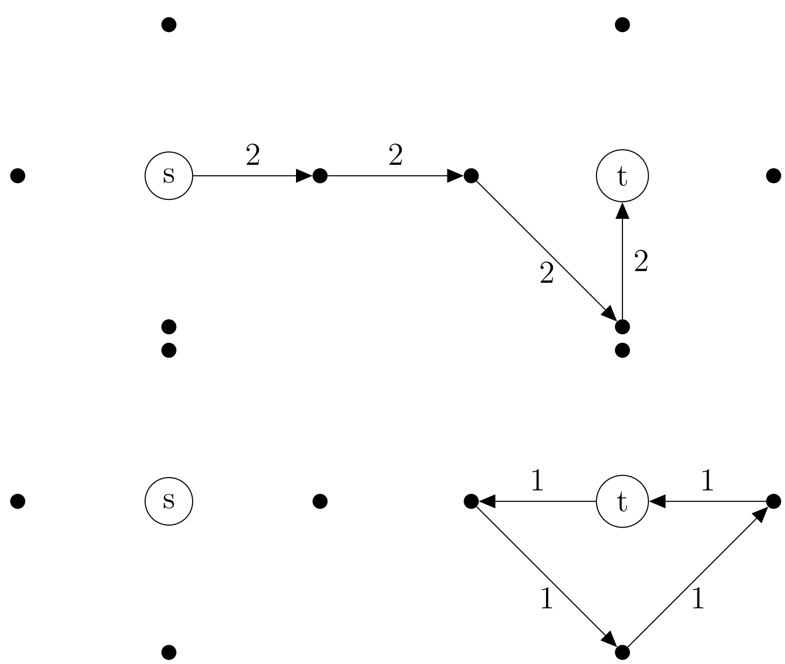
$$y_s = 0, y_1 = -3, y_2 = -4, y_3 = -8, y_4 = -10, y_5 = -6, y_6 = -5, y_t = -11$$

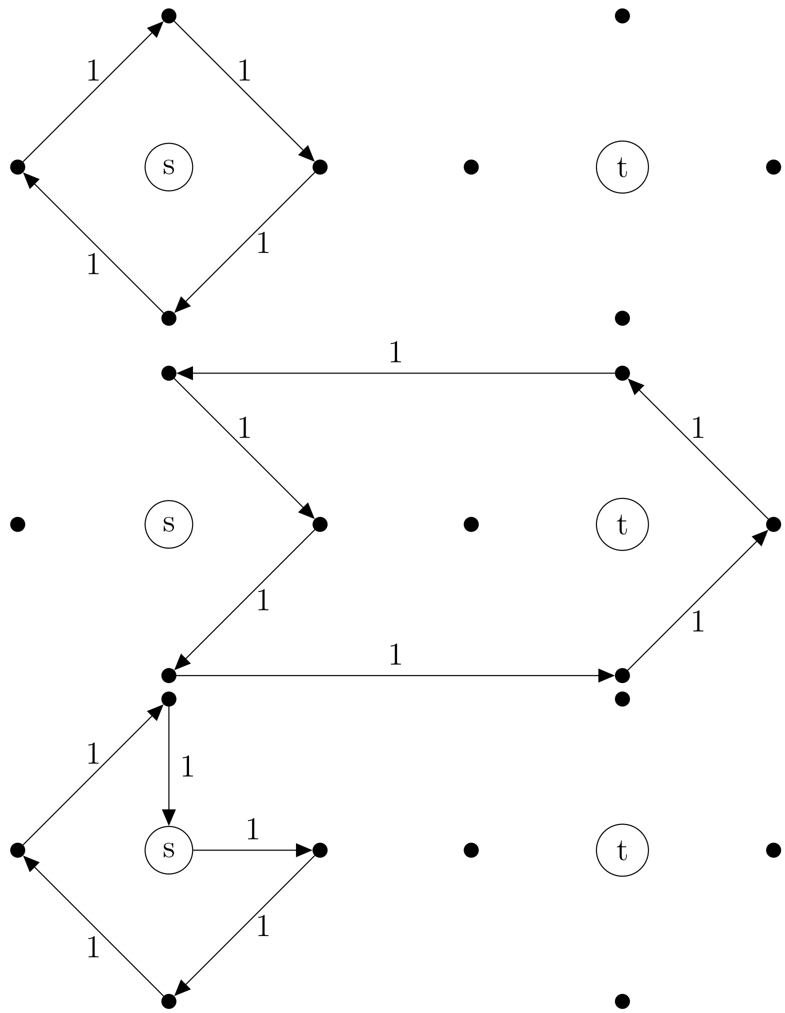
Again, the value of the solution is 11, which proves the optimality of the shortest path.

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.

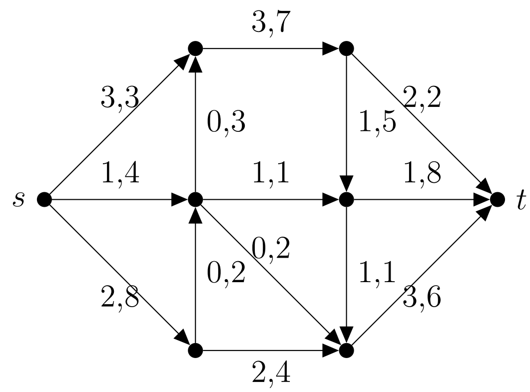


Solution:
Here is a possible decomposition:



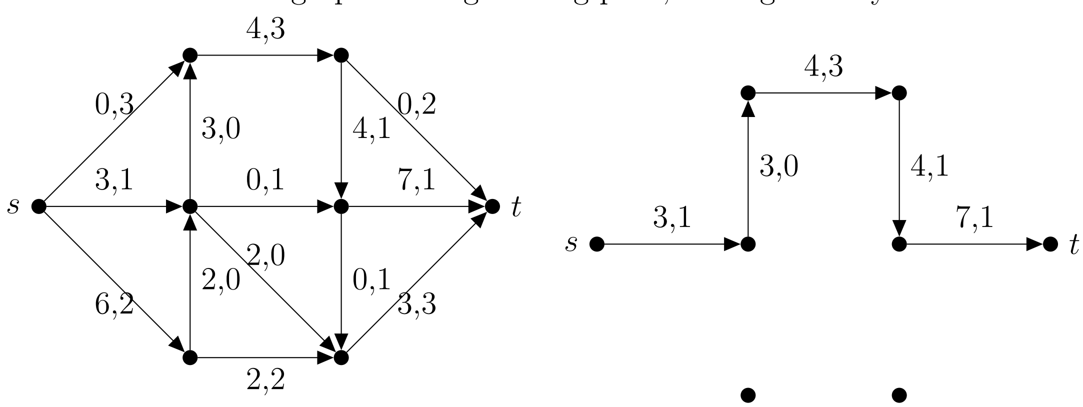


3: (*Augmenting paths*)
Consider the network below with given capacity and flow values. (The edge label f, u means flow-value f and capacity u.) Find augmenting paths and augment the flow to a maximum flow. Provide the list of residual graphs AND augmenting paths. It other words, run Ford-Fulkerson algorithm.

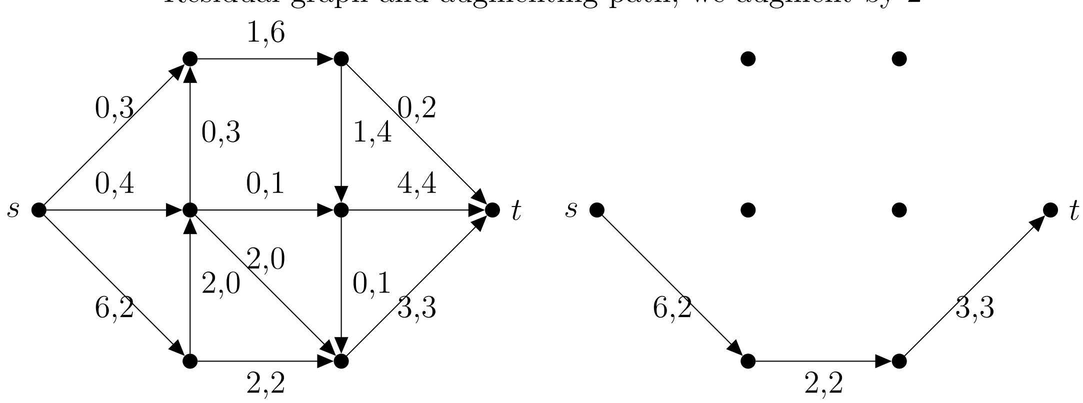


Solution:
First we build a residual graph. In order to simplify the drawing, every edge is assigned 2 numbers. The first number is residual capacity in the direction of the edge, second number is the residual capacity in the opposite direction. Note that the original flow is the second number and the difference between capacity an the original flow is the first number - in other words, somebody was lazy to draw arrows in both directions and make proper figures.

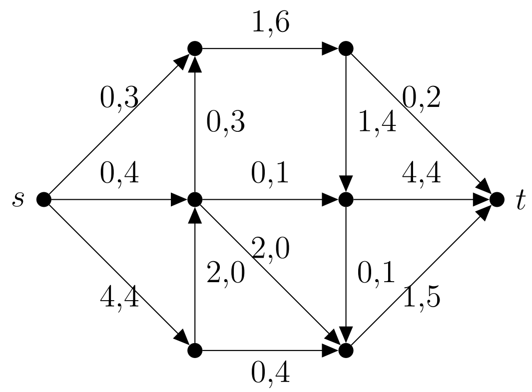
Residual grap and Augmenting path, we augment by 3:

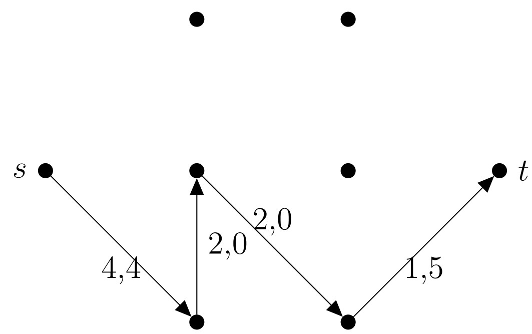


Residual graph and augmenting path, we augment by 2

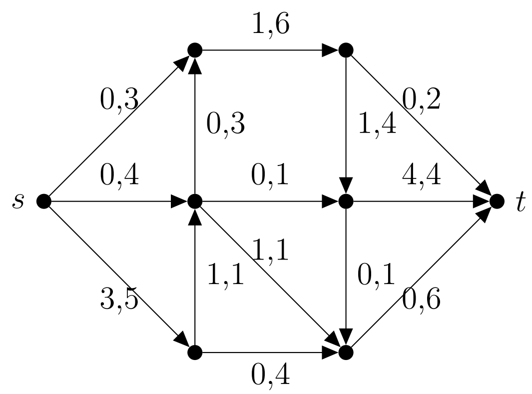


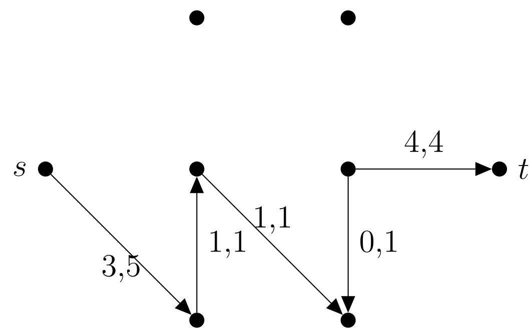
Residual graph and augmenting path, we augment by 1



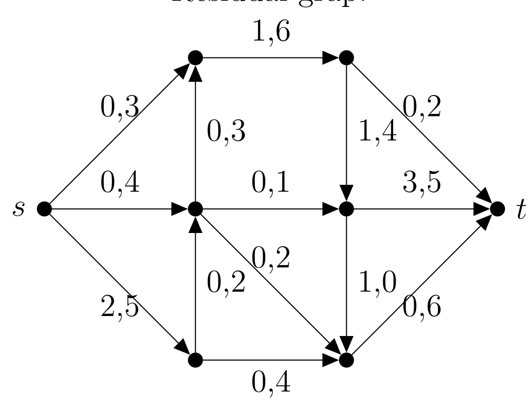


Residual grap and augmenting path, we augment by 1

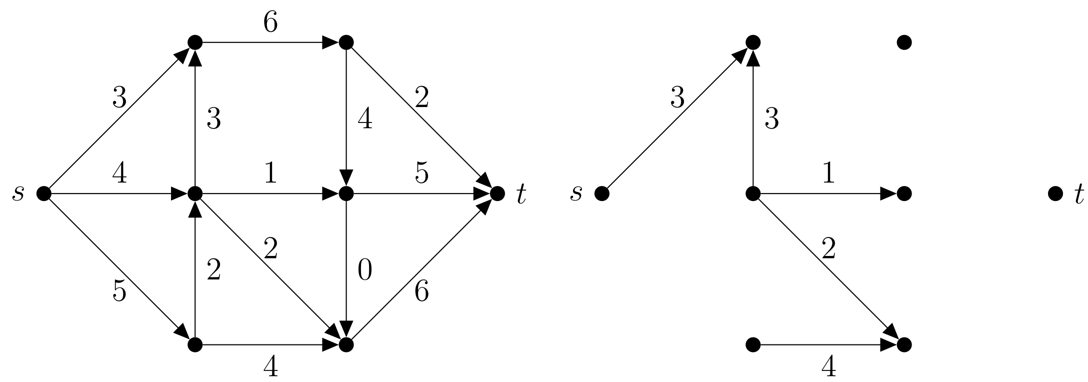




Residual grap:



This gives graphs of the maximum flow and minimum cut:



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).

Solution:
First observe that

$$\delta^+(X \cap Y) + \delta^+(X \cup Y) \leq \delta^+(X) + \delta^+(Y).$$

This can be seen by considering edges that leave X and/or Y. Since $\delta^+(X)$ and $\delta^+(Y)$ are minimum cuts, we have $\delta^+(X) = \delta^+(Y) \leq \delta^+(X \cap Y)$ and $\delta^+(X) = \delta^+(Y) \leq \delta + (X \cup Y)$. Hence we obtain

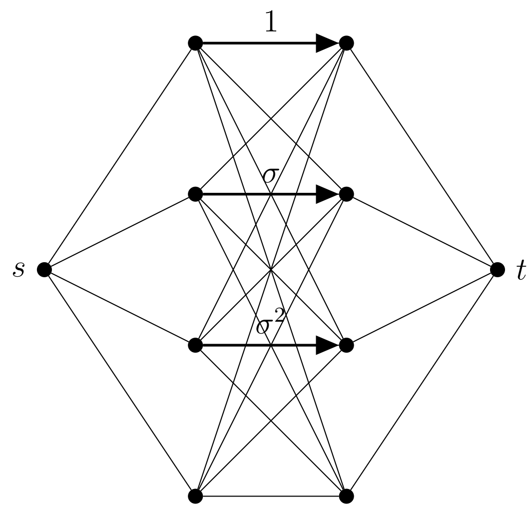
$$\delta^+(X) + \delta^+(Y) \leq \delta^+(X \cap Y) + \delta^+(X \cup Y) \leq \delta^+(X) + \delta^+(Y).$$

Hence all $\leq$ must be $=$ and we have

$$\delta^+(X) = \delta^+(Y) = \delta^+(X \cap Y) = \delta + (X \cup Y).$$

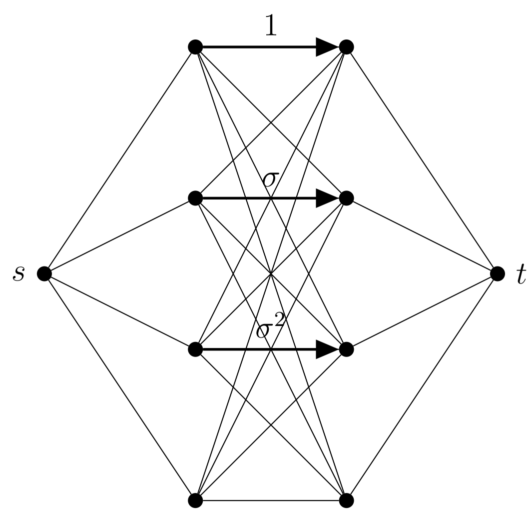
Therefore $X \cap Y$ and $X \cup Y$ are also minimum s-t-cuts.

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.

Solution:
Let $\sigma = \frac{\sqrt{5}-1}{2}$. Note that σ satisfies $\sigma^n = \sigma^{n+1} + \sigma^{n+2}$ and $\sum_i \sigma^i = \frac{1}{1-\sigma} = S$. Consider a graph, where capacities of edges are S except three edges with capacities 1, σ, σ^2. See the diagram.



Maximum flow is clearly $3S$. We try to fool Ford-Fulkerson algorithm by using paths that do improvements of the order σ^i. We use edges 1, σ and σ^2 to construct paths that can improve the flow by at most σ^i. Here is example of fooling choices. The values under 1, σ, σ^2 correspond to the residual capacity in the direction from s to t as depicted in the figure. In augmenting path, we list in which direction we traverse the edges 1, σ, σ^2. The remaining edges always have enough remaining capacity.

#	augmenting path	flow value	1	σ	σ^2
0			1	σ	σ^2
1	$\overrightarrow{1}$	1	0	σ	σ^2
2	$\overrightarrow{\sigma^2}, \overrightarrow{\sigma}, \overleftarrow{1}$	σ^2	σ^2	σ^3	0
3	$\overrightarrow{\sigma}, \overrightarrow{1}, \overleftarrow{\sigma^2}$	σ^3	σ^4	0	σ^3
4	$\overrightarrow{1}, \overrightarrow{\sigma^2}, \overleftarrow{\sigma}$	σ^4	0	σ^4	σ^5

We can show by induction how the successive step are decreasing in the flow value. If we have three edges with residual capacities σ^i, σ^{i+1} and 0, we can traverse the edges σ^i, σ^{i+1} in forward direction and edge 0 in backward direction. The maximum possible augmentation is σ^{i+1}. New residual capacities will be $\sigma^i - \sigma^{i+1} = \sigma^{i+2}$, $\sigma^{i+1} - \sigma^{i+1} = 0$ and $0 + \sigma^{i+1} = \sigma^{i+1}$. This leaves is with the same setting but i increased by 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.

  Solution:

  Suppose T contain a red edge $uv = e$. Since e is red, there is cycle C where e is heaviest. Notice that $T - e$ is disconnected, but $C - e$ is a $u - v$ path that contains an edge f such that $T - e + f$ is connected. Since $w(f) < w(e)$, T was not MST.

- Show that blue edge must be in MST.

  Solution:

  Suppose T does not contain a blue edge $uv = e$. Since e is red, there is cut X where e is the lightest. Since there is a $u - v$ path in T, there must be $f \in E(T) \cap X$. Since $w(e) < w(f)$, $T - f + e$ has smaller weight than T. Hence T was not MST.

- Show that blue edges form a tree

  Solution:

  Blue edges cannot form a cycle, otherwise the heaviest edge on the cycle should be red. If blue edges do not form a connected graph, then it will give a cut without blue edges but every cut contains a blue edge.

- Show that every edge gets colored.

  Solution:

  Suppose e is not colored. If it is in any cut that has no blue edges, one more edge can be made blue. Hence all cuts containing e contain also a blue edge.

- Show that no edge satisfies both red and blue criteria. (i.e. every edge has one color).

  Solution:

  Suppose $uv = e \in E(G)$ is both blue and red. Since it is red it contains a cycle C where it is heaviest. Since it is blue it is in a cut X that where it is lightest. But C and X must intersect at e and at least one other edge f. But e red means $w(f) < w(e)$ and e blue means $w(e) < w(f)$ which is a contradiction.

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.