CMPS 6610 Algorithms – Fall 16

11/15/16

8. Homework

Due **11/22/16** at the beginning of class

1. **Floyd-Warshall (9 points)**
   (a) (3 points) Show how to use Floyd Warshall's algorithm to detect whether a weighted graph contains a negative weight cycle. Justify your answer.
   (b) (3 points) Show that Floyd-Warshall's algorithm can be implemented to use only $\Theta(n^2)$ space, by simply dropping all the superscripts. Argue that this algorithm correctly computes all shortest path weights for all pairs of vertices. (See problem 25.2-4 on page 699 in the book.)
   (c) (3 points) In Floyd Warshall's algorithm, can you switch the order of the three for-loops and still compute the correct output?

2. **Transitive Closure (6 points)**
   Let $G = (V, E)$ be a directed graph and let $G^* = (V, E^*)$ be its transitive closure. Assume G^* has been computed and is represented using an adjacency matrix.
   (a) (3 points) Show how G^* can be updated in $O(|V|^2)$ time when one edge is added to G.
   (b) (3 points) Give an example of a graph G and an edge e to be inserted, such that updating G^* takes $\Omega(n^2)$ time, no matter what algorithm is used.

3. **Ford-Fulkerson (5 points)**
   Run Ford-Fulkerson's algorithm on the flow network on the next page. When choosing augmenting paths, use at least one non-shortest path.

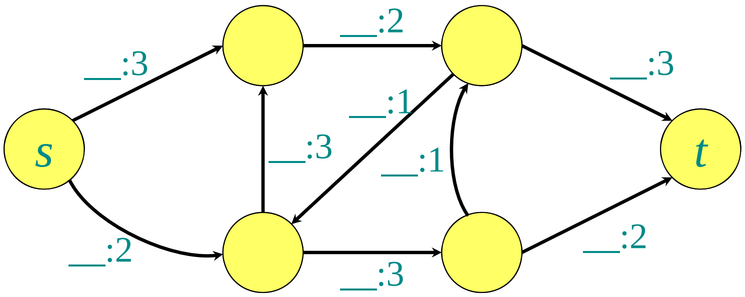
   Start with a zero flow on all edges and initialize the residual network. Then repeatedly find an augmenting path in the residual network (draw it in), increase the flow and update the residual network.

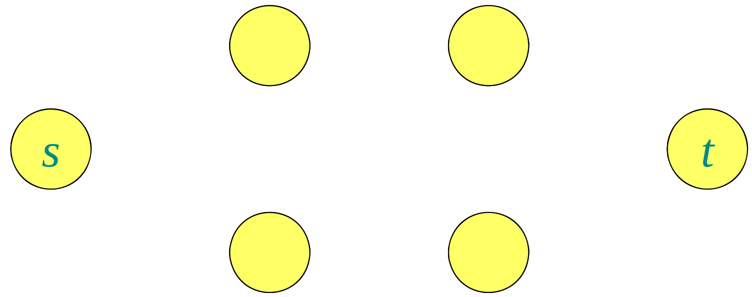
4. **Updating Flow (6 points)**
   Suppose you are given a flow network $G = (V, E)$ with source s, sink t, and non-negative integer capacities c. You are also given a maximum flow f on G.

   We would like to be able to update the maximum flow efficiently when the capacity of an edge $e \in E$ is incremented by 1 or decremented by 1. Give **efficient** algorithms for INCREMENT(e) and for DECREMENT(e) and analyze their complexities. Your algorithms should be faster than simply recomputing the maximum flow.

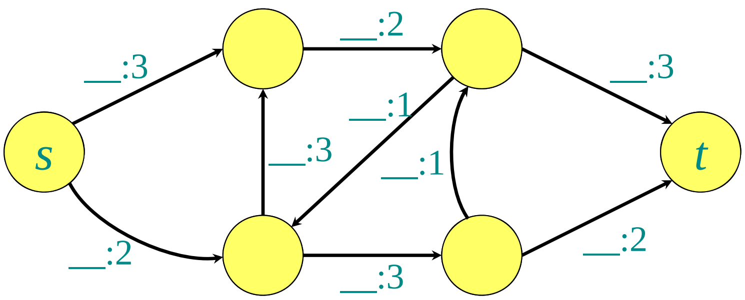
FLIP OVER TO BACK PAGE $\Longrightarrow$

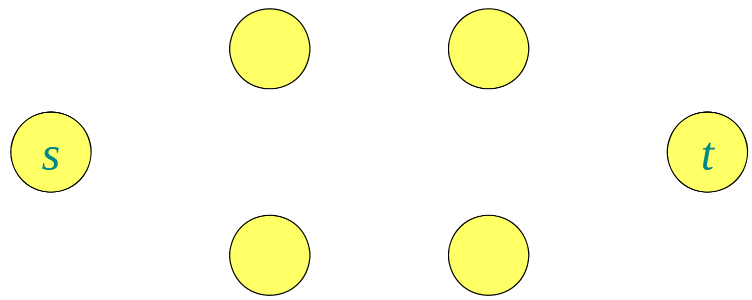
Network **Residual network**



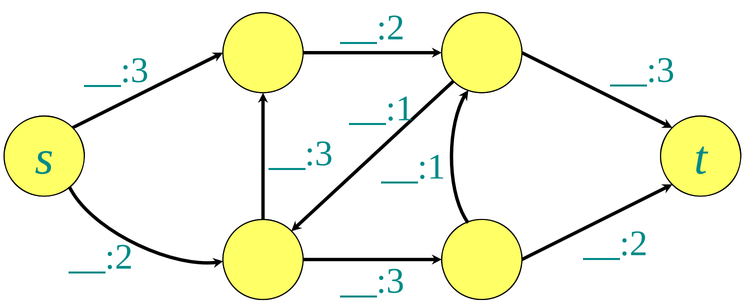


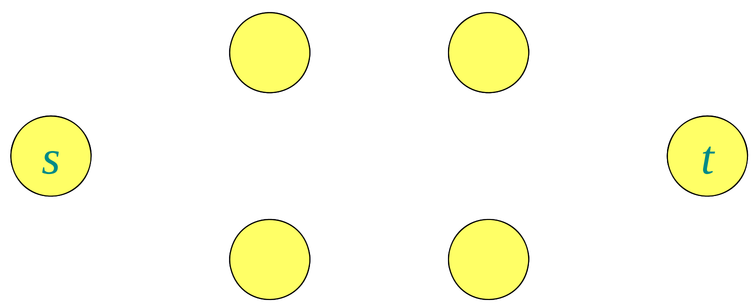
Increase flow value by: ____



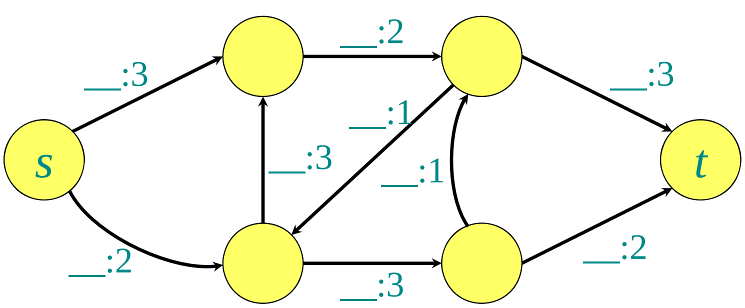


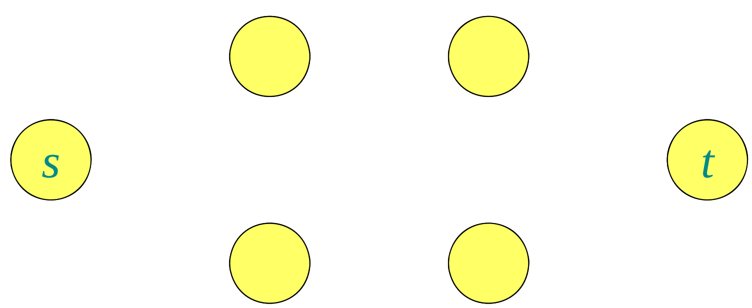
Increase flow value by: ____





Increase flow value by: ____





Increase flow value by: ____