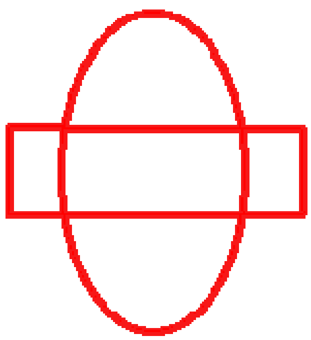
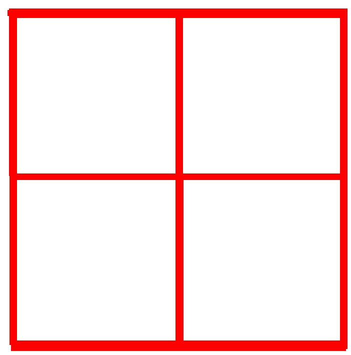
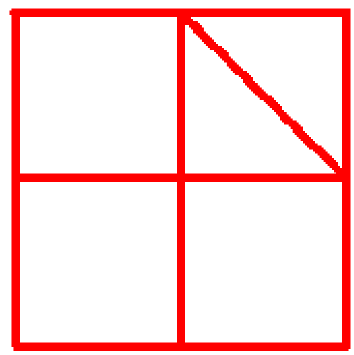
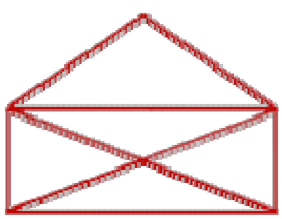
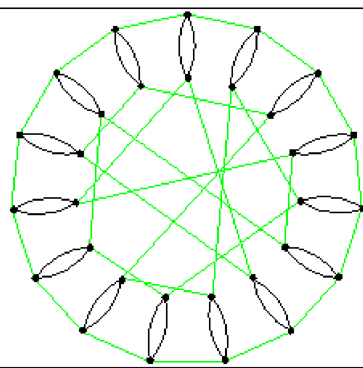
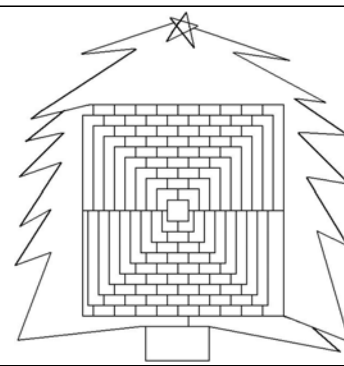
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
15.053 – Optimization Methods in Management Science (Spring 2007)

Recitation 6, April 5th and April 6th, 2007

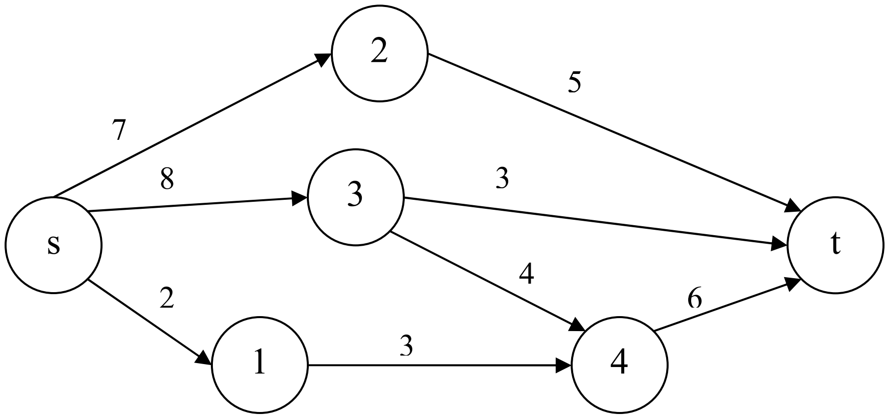
Problem 1 (*Eulerian Graphs*)

Which of the graphs below have Eulerian cycles/paths?

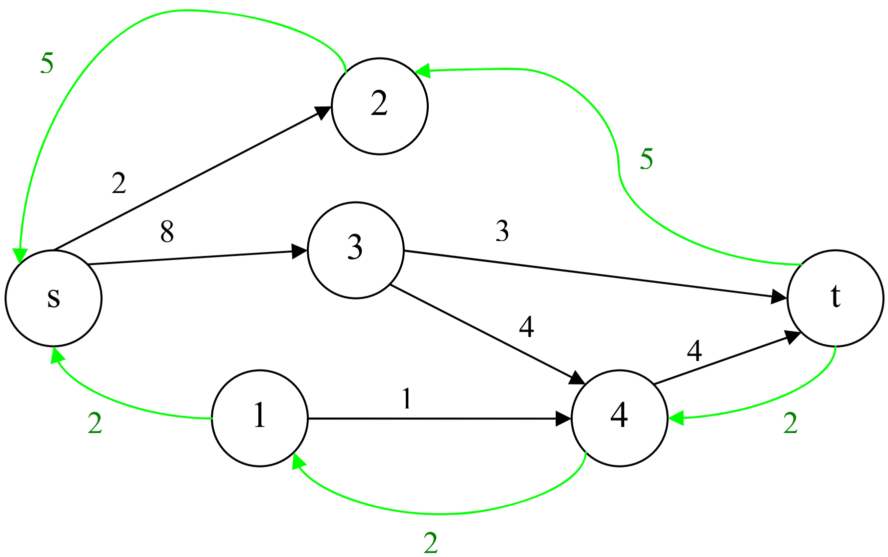
		
		

Problem 2: Ford-Fulkerson Algorithm

Consider the following Maximum Flow network.



And consider the following residual network resulting from two iterations of the Ford-Fulkerson algorithm.



Part A:

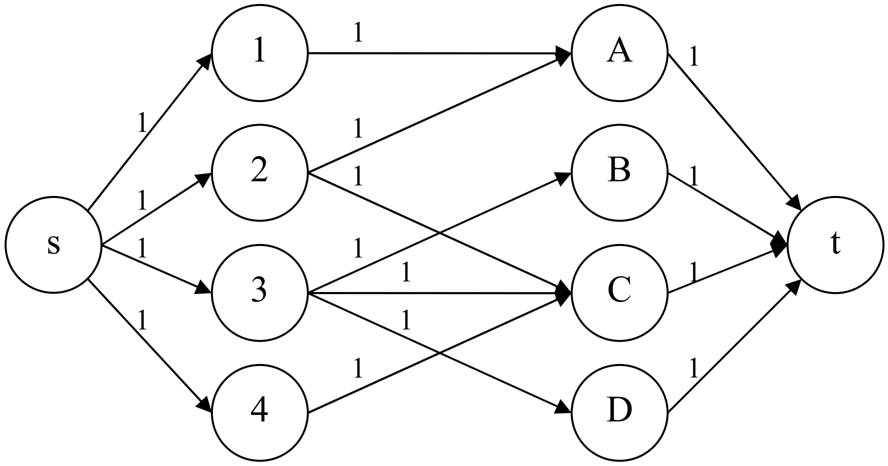
Perform the final two iterations of the Ford-Fulkerson algorithm to determine the maximum flow.

Part B:

Determine the corresponding minimum cut based on the final residual network.

Problem 3 (*Max Flow*)

For parts A through C, please consider the following Maximum Flow Problem, which represents a matching problem. All arcs have capacity 1.



To achieve a maximum flow of 3 using the Ford-Fulkerson algorithm, we choose the following three augmenting paths, s→1→A→t, s→2→C→t, and s→3→B→t.

Part A

Draw the final residual network based on this choice of augmenting paths.

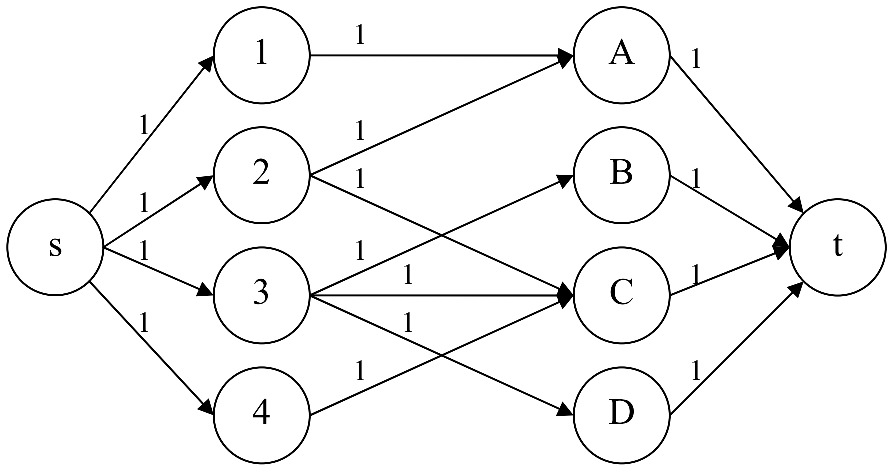
Part B

Determine a minimum s-t cut for this problem.

Part C

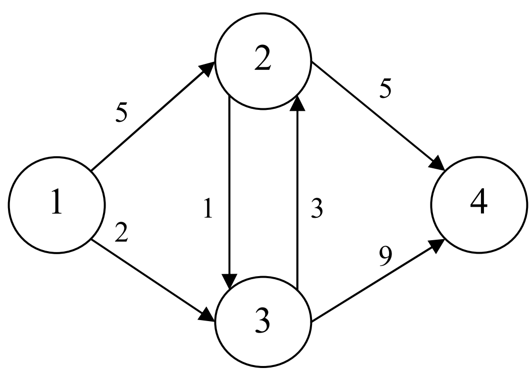
Update the network diagram below to translate the Maximum Flow Problem into a Minimum Cost Network Flow Problem. Please be sure to clearly specify all arc capacities and costs.

Draw all the necessary additional components.



Problem 4 (*Shortest Path)*

Consider the following network with path lengths shown



Iteration 1:

Node	1*	2	3	4
Distance	0	5	2	∞
Predecessor		1	1	-

Use Dijkstra's algorithm to find the shortest path from node 1 to all other nodes. The results of the first iteration of Dijkstra's algorithm are provided in the table below the network. Please fill in the appropriate values for the subsequent iterations in the provided tables. Put a* next to the node headers to denote which values are permanent (e.g., node 1 is permanent after the first iteration).

Iteration 2:

Node	1*	2	3	4
Distance	0			
Predecessor				

Iteration 3:

Node	1*	2	3	4
Distance	0			
Predecessor				

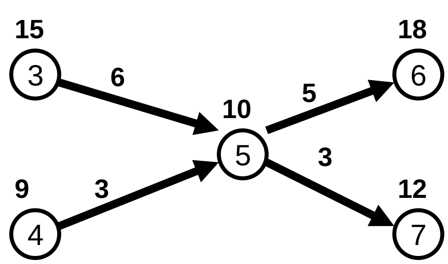
Iteration 4:

Node	1*	2	3	4
Distance	0			
Predecessor				

Problem 5 ***Shortest Path***

Consider the part of the directed graph *G = (N, A)* shown in the figure below. The other arcs and nodes in the network are not shown. The numbers on the arcs are the arc lengths.

Dijkstra's algorithm is finding the shortest path from node s to node t, neither of which are shown. Suppose node *5* is made permanent at the current iteration. The numbers above the nodes are the distance labels created by Dijkstra's Algorithm in previous iterations.



Part A

Which of nodes *3*, *4*, *6*, and *7* are not yet permanently labeled? Which of the nodes are permanently labeled? If you do not have enough information to determine the answer, then say so.

Part B

Does node *5* have a predecessor (as used in Dijkstra's algorithm)? If you answered yes, can it be one of the nodes *3*, *4*, *6* or *7*? (if you answered no, you can skip this question)

Part C

After node *5* is made permanent, Dijkstra's algorithm performs update(5), which leads to arcs being scanned and distances being updated. What are the distance labels of nodes *3*, *4*, *6*, and *7* after this update is performed?

Problem 6 *(True or False)*

For A-D suppose Turkey Tim is running Dijkstra's algorithm and he accidentally selects the node on LIST with the second lowest distance label rather than the one with minimum distance label at some iteration. Answer the following questions true or false:

Part A

It is possible the algorithm will terminate with an incorrect shortest path.

Part B

It is possible the algorithm will terminate with the correct shortest path.

Part C

It is possible the algorithm will not terminate.

Part D

It is possible the algorithm will terminate in less iteration than if the error was not made.

Part E

For a Minimum Cost Network Flow Problem, a supply of -1 for a node is equivalent to a demand of 1.

Part F

For a Minimum Cost Network Flow problem, if the specified supply is not equal to the demand (that is, the sum of the b_i's does not equal 0), the corresponding Linear Program with flow balance equality constraints could still be feasible.

Part G

If we add a directed arc from s to t of capacity three, it is possible the max flow will increase by less than three

Part H

It is possible for an undirected graph to have an Eulerian Path but not an Eulerian Cycle.

Part I

Suppose all arcs in a network have different lengths. Then the network has a unique shortest path.

Part J

If we delete all arcs on an s-t path from a graph, the min cut value must strictly decrease (assume all arcs have positive capacity).