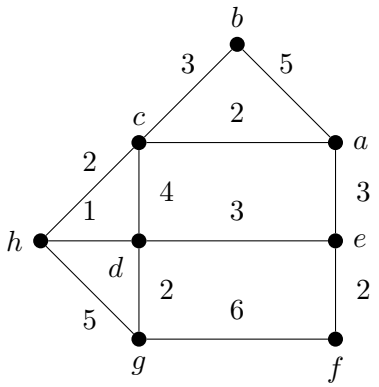
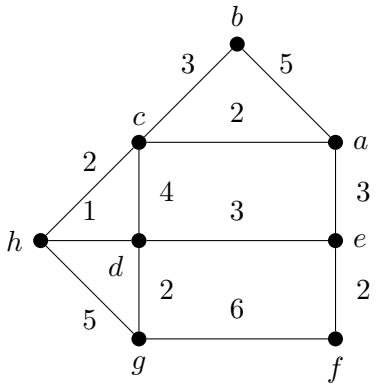
Caleb Logemann
MATH 566 Discrete Optimization
Homework 7

1. A cut of G is *minimal* if there is no cut of G properly contained in it. Prove that the random contraction algorithm returns only minimal cuts.

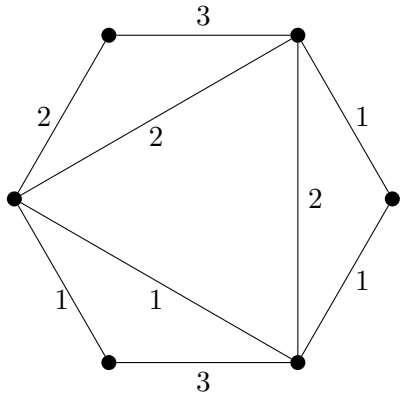
2. Implement Node Identification Minimum Cut Algorithm. Try it on the following graph:



3. Implement Random Contraction Algorithm. Try it on the following graph. Run it many times and based on your experiment conclude what is the probability that that your algorithm succeeds on this graph.

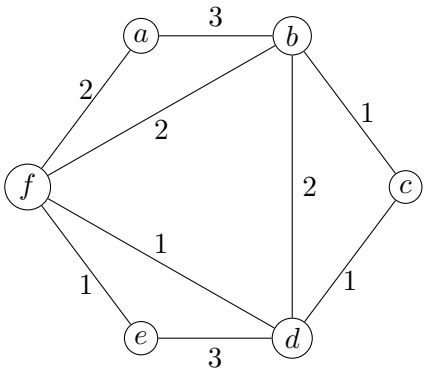


4. Construct Gomory-Hu Tree for the following graph using the algorithm from the class. Numbers on edges correspond to capacities. Show Show steps after every new cut.

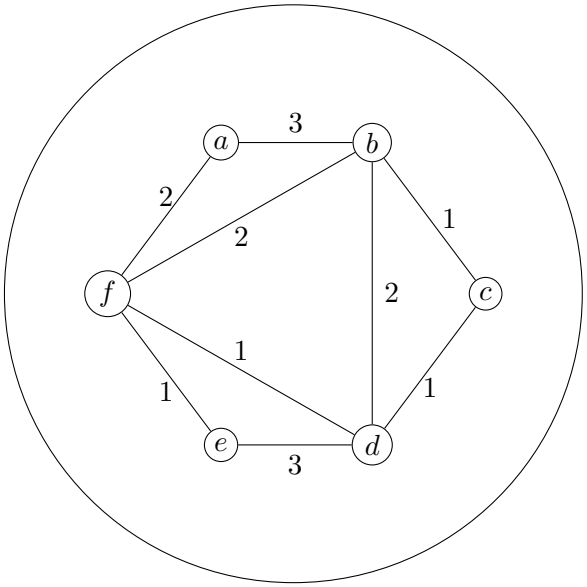


Check your answer with Sage using method `Graph.gomory_hu_tree()`. Do not implement it yourself, just run it.

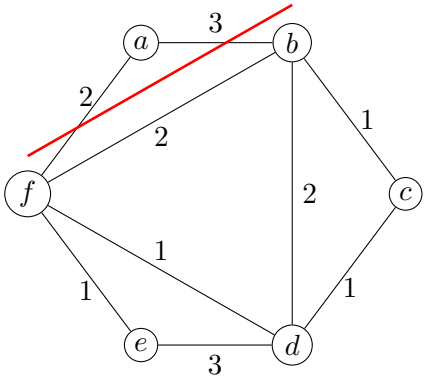
First in order to run the Gomory Hu algorithm, I will label the vertices.



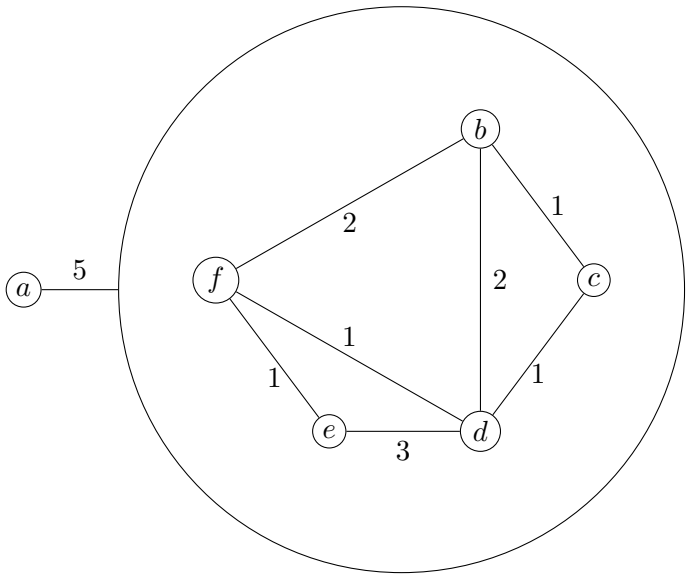
First start with a tree that is $T = (\{V\}, \emptyset)$. This tree looks like.



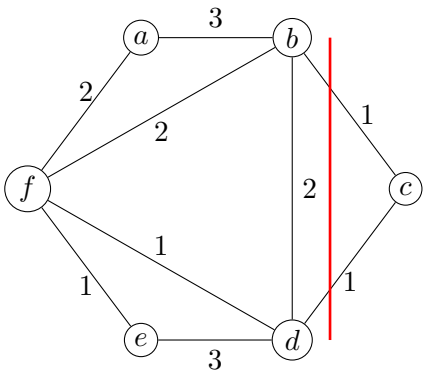
Now I will chose to find the minimum a-b cut, which corresponds to finding the minimum a-b flow in the following graph. The minimum cut found is shown in red.



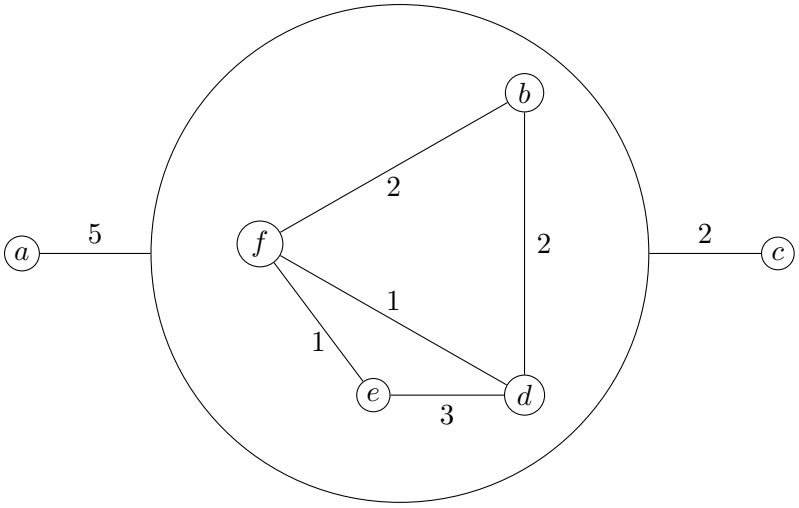
Using this cut the new Gomory Hu tree is



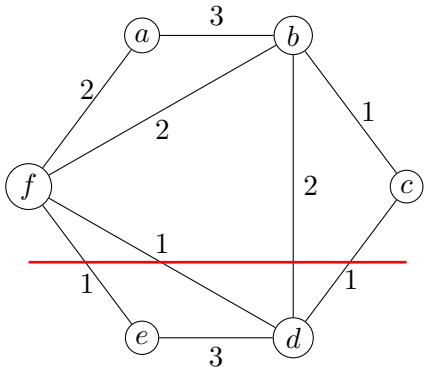
Now I will find the minimum b-c cut in the following graph. The minimum cut found is shown in red.



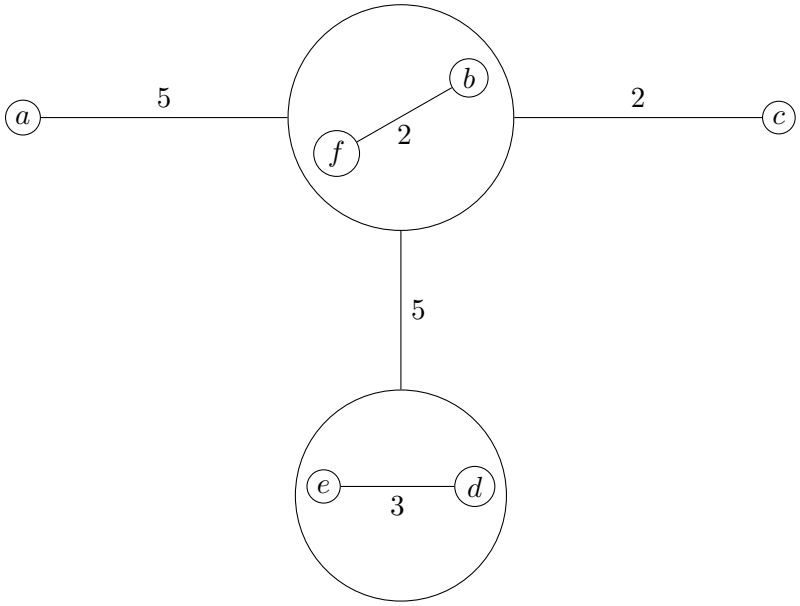
Using this cut the new Gomory Hu tree is



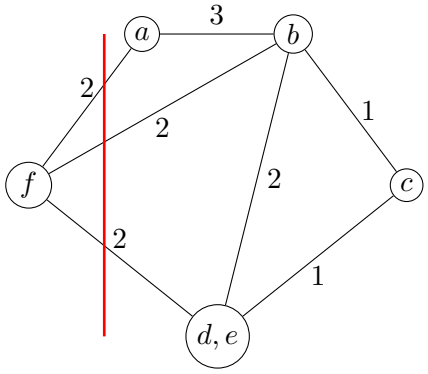
Now I will find the minimum b-d cut in the following graph. The minimum cut found is shown in red.



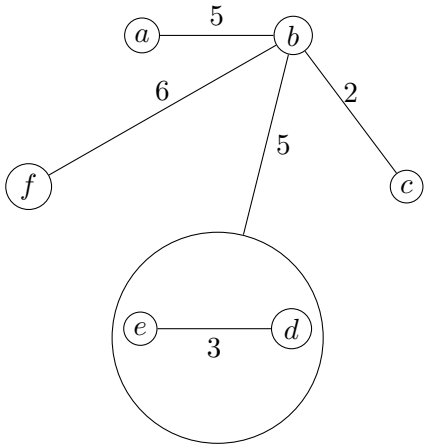
Using this cut the new Gomory Hu tree is shown below.



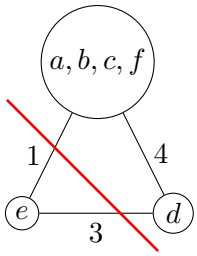
Now I will find the minimum b-f cut in the following graph, where e and d are contracted.



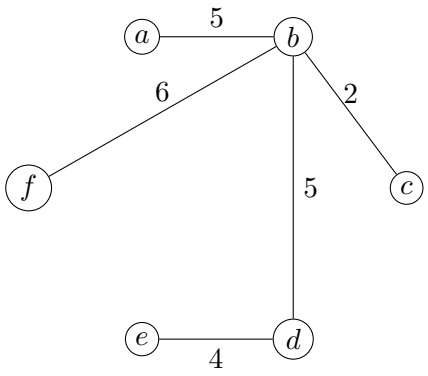
Using this cut the new Gomory Hu graph is



Lastly I will find the minimum d-e cut in the following contracted graph.



Thus the full Gomory Hu tree is



I checked this in Sage with the following script.

```
m = matrix([
     (0, 3, 0, 0, 0 ,2),
     (3, 0, 1, 2, 0, 2),
     (0, 1, 0, 1, 0, 0),
     (0, 2, 1, 0, 3, 1),
     (0, 0, 0, 3, 0, 1),
     (2, 2, 0, 1, 1, 0)])
graph = Graph(m, weighted=True)
graph.relabel({0:'a', 1:'b', 2:'c', 3:'d', 4:'e', 5:'f'})
gomory_hu_tree = graph.gomory_hu_tree()
gomory_hu_tree.plot(edge_labels=True, layout='tree').show()
```

The output of this script is the following image of the Gomory Hu tree.