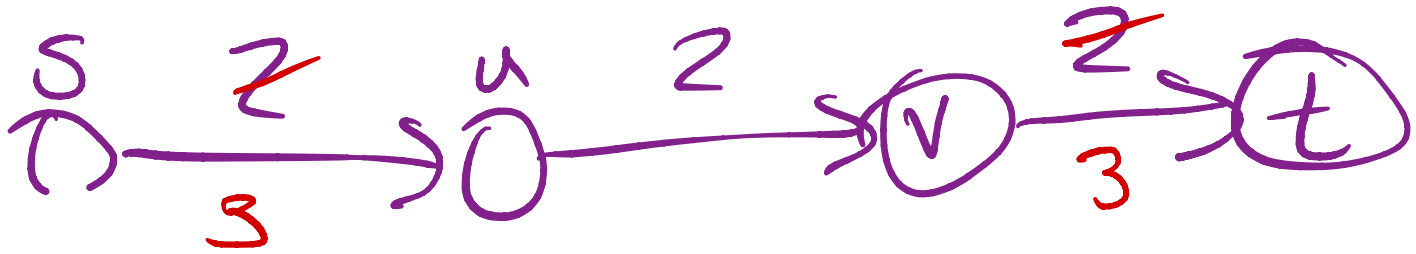
Topics:

- Huffman Coding
- Graph Search
- BFS, DFS
- Minimum Spanning Trees
- Generic MST Algorithm
- Prim's & Kruskal's Algorithms
- Max Flow, Min Cut
- Ford-Fulkerson Algorithm
- Bipartite Matching
- Dynamic Programming
- Fibonacci numbers
- Bellman-Ford Algorithm
- Dijkstra's Algorithm
- Knapsack

Let $(S, V - S)$ be a minimum (s, t)-cut of the flow graph G. Let (u, v) be an edge that crosses the cut in the forward direction i.e. $u \in S$ and $v \in V - S$. Increasing the capacity of this edge always increases the maximum flow of G.

A. True

B. False



If we have a dynamic programming algorithm with n^2 subproblems, is it possible for the runtime to be asymptotically larger than $\Theta(n^2)$?

A. Yes

B. No

if $\Theta(n)$ to solve one subproblem

Consider the knapsack problem with $A = [(9,1), (3,5), (4,9), (8,2), (10,9), (2,5)]$ where each pair is (value, weight). Determine the optimal set of items which maximizes the total value if $W = 12$ is the weight limit. Fill in the DP table.

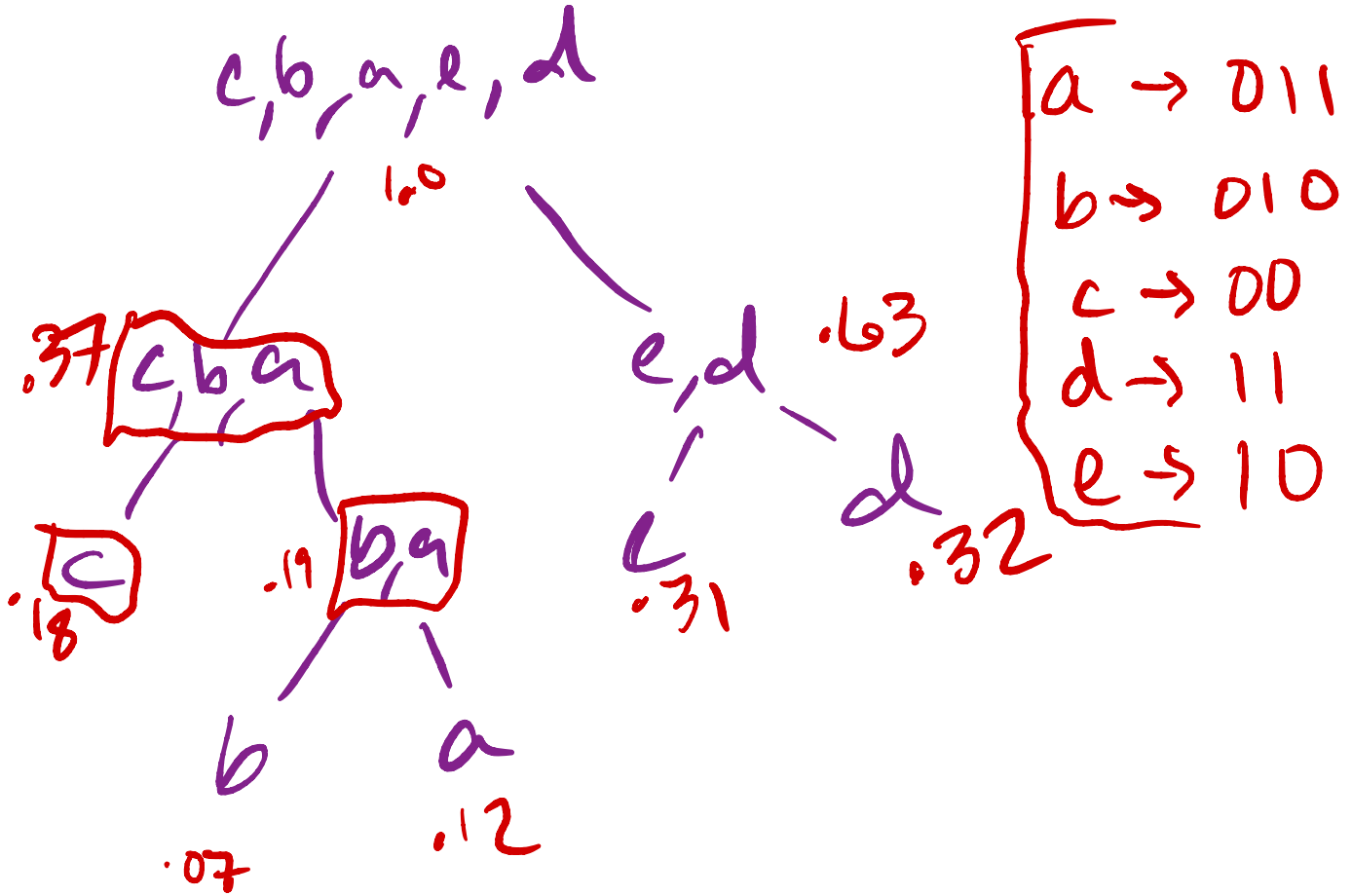
	0	1	2	3	4	5	6	7	8	9	10	11	12
0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	9	9	9	9	9	9	9	9	9	9	9	9
2	0	9	9	9	9	9	12	12	12	12	12	12	12
3	0	9	9	9	9	9	12	12	12	12	13	13	13
4	0	9	9	17	17	17	17	17	20	20	20	20	21
5	0	9	9	17	17	17	17	17	20	20	20	20	27
6	0	9	9	17	17	17	17	17	20	20	20	20	27

cells alg. visits

• item in solution

Given the following character frequencies, construct the Huffman code. Assume traversing the binary tree to the left indicates a 0 and to the right is a 1.

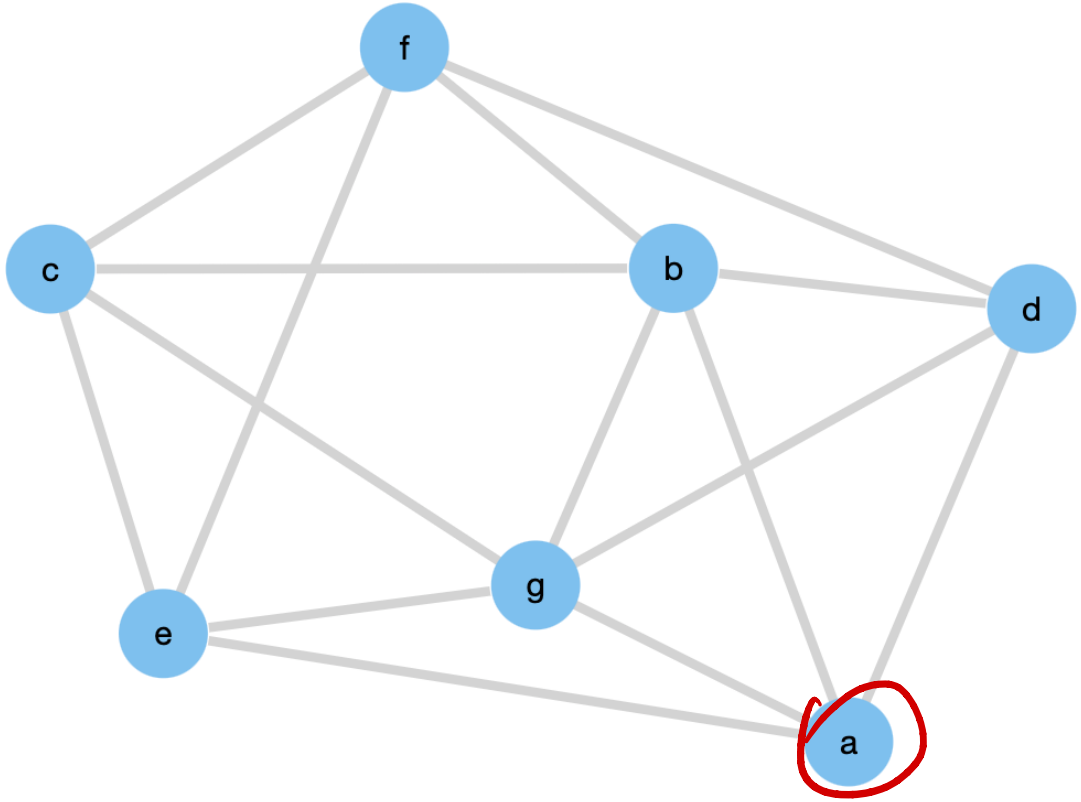
a	0.12
b	0.07
c	0.18
d	0.32
e	0.31

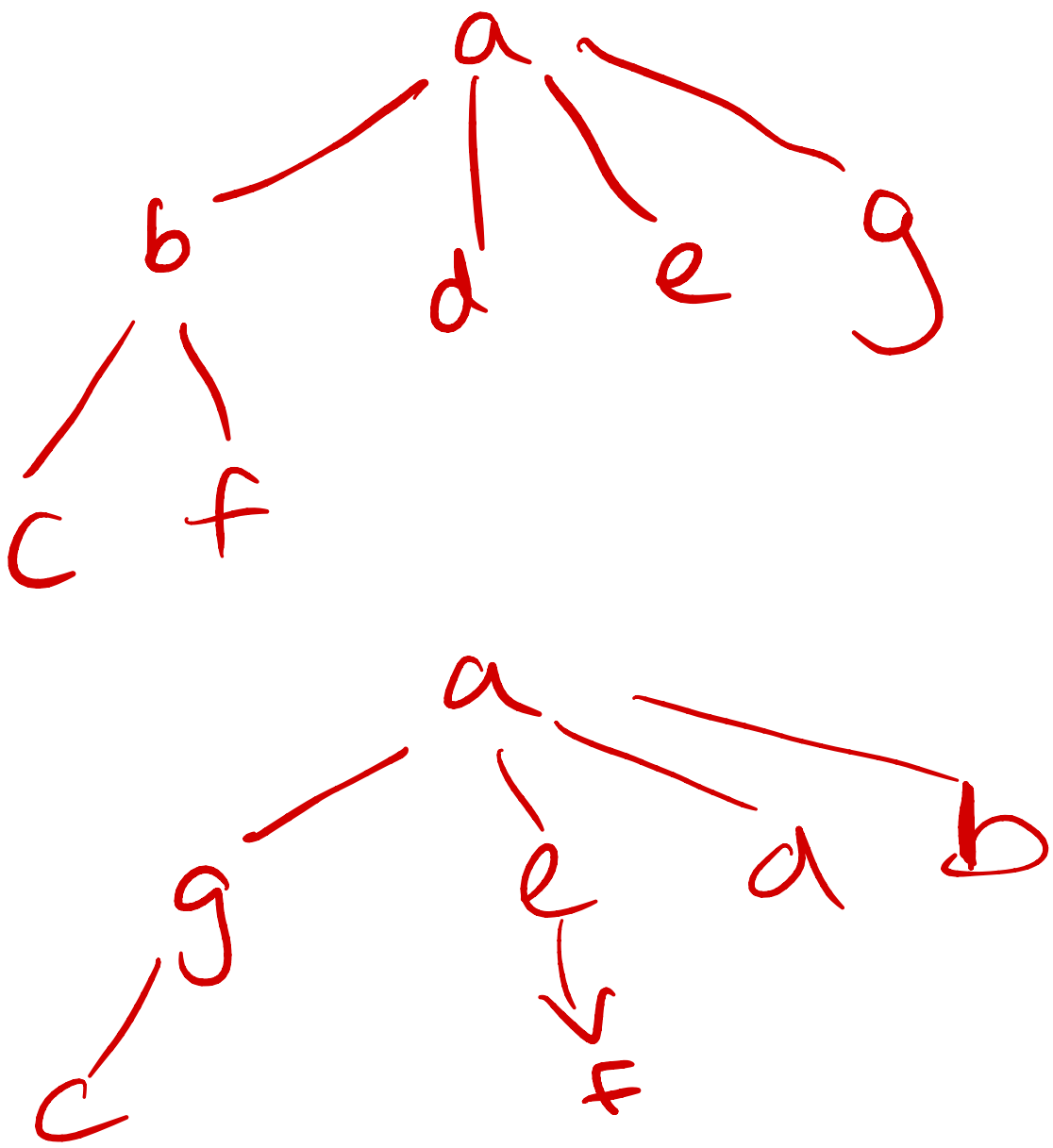


[b, a, c, e, d]

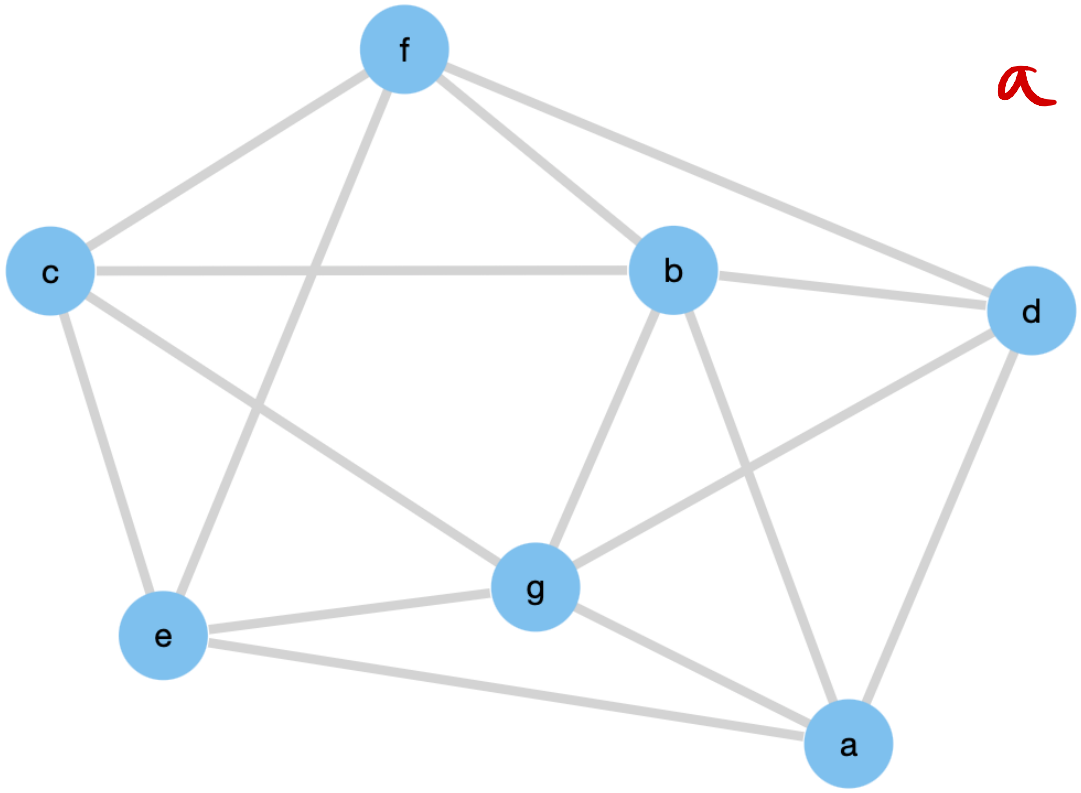
[c, (a,b), e, d]

Draw the BFS tree of the following graph starting from node a



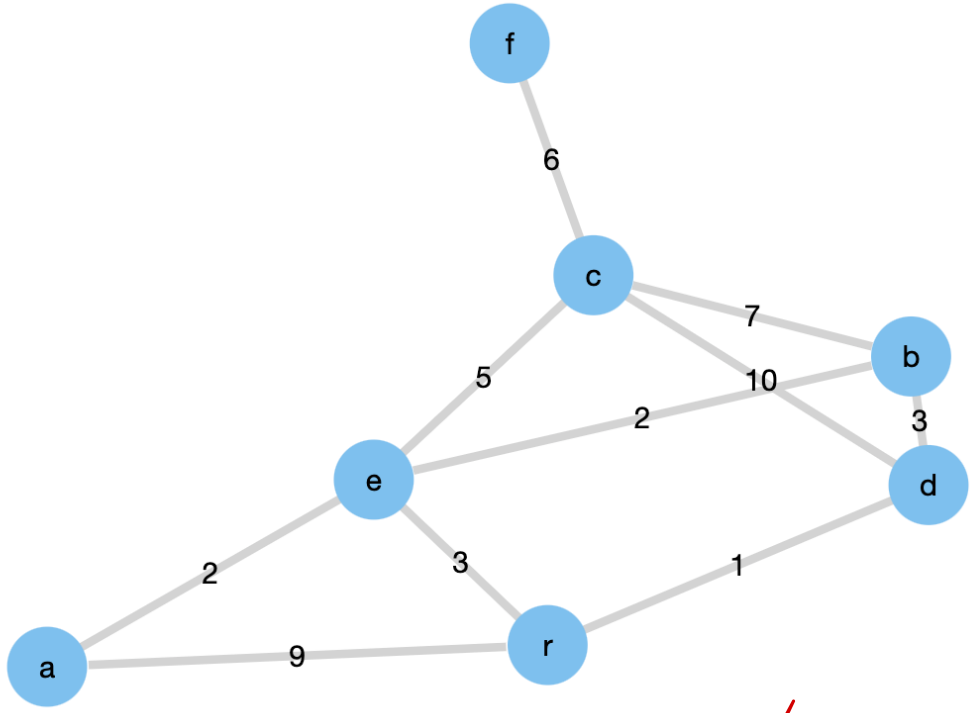


Draw the DFS tree of the following graph starting from node a



a → b → c → e → f → d → g

Compute the SSSP from r to all other nodes in the network below using Dijsktra's algorithm.



a 5
b 4
c 8
d 1
e 3
f 14

D = 5 4 8 1 3 14

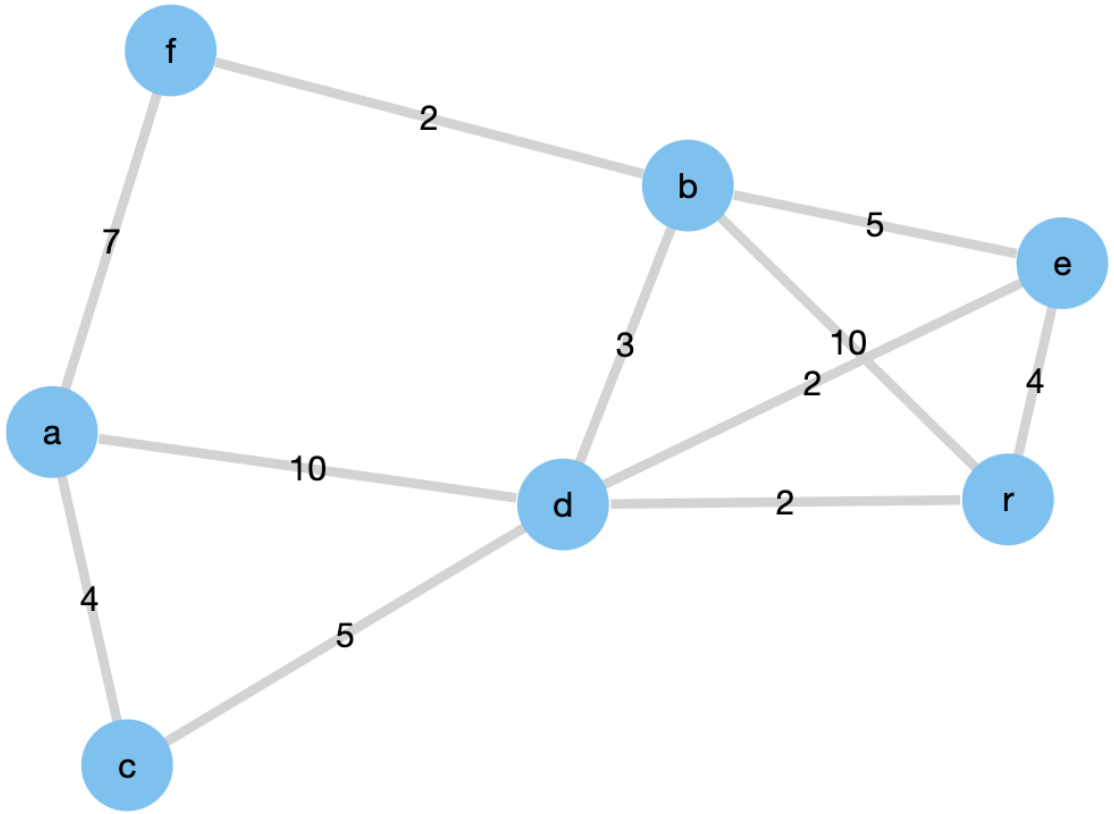
f 0

r

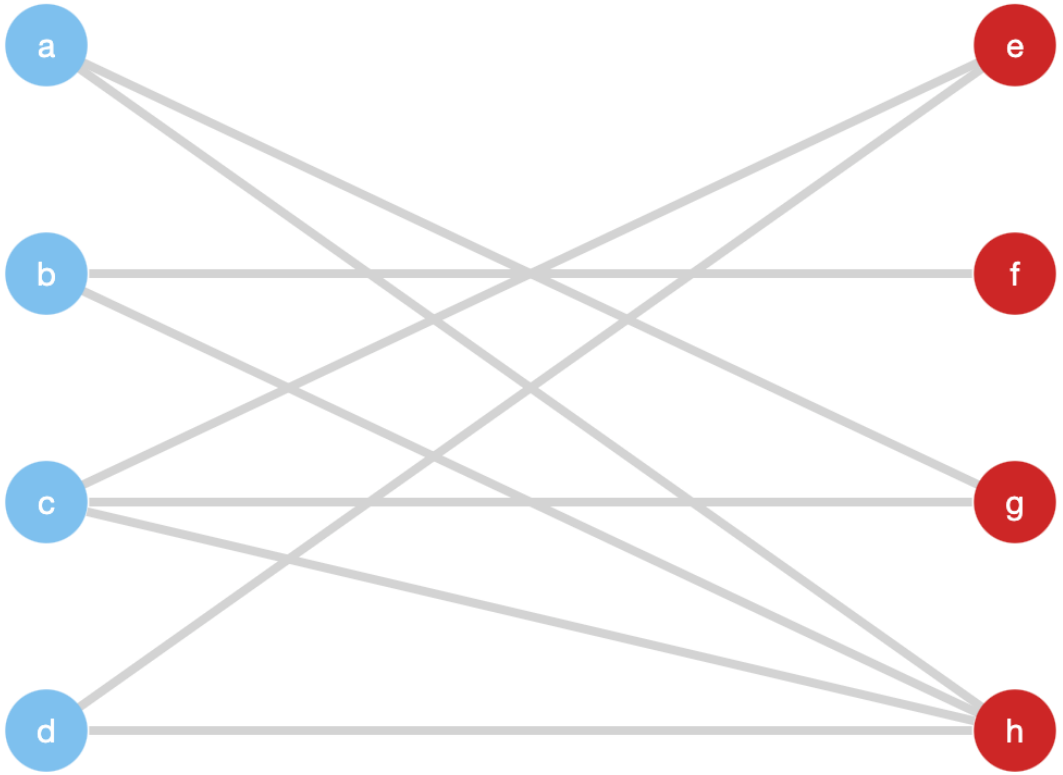
Q = {a, b, c}

{d, e, b, a, c, f}

Compute the SSSP from r to all other nodes in the network below using Bellman-Ford's algorithm.



Determine the maximum bipartite matching of the following bipartite graph.



Determine the maximum bipartite matching of the following bipartite graph.