CSE 551: Quiz 6 Solutions

Jamison Weber

October 23, 2021

Note that some of the solutions provided in this document may apply to questions that did not appear in your particular quiz.

1 Problem 1

Suppose we are given a network partitioned into sets A and B such that source node $s \in A$ and sink node $t \in B$. This partition creates a cut in the network. Assume the edges that traverse the cut have an infinite capacity, and the edge values given represent the respective flow along each corresponding edge. Determine the total flow of this network.

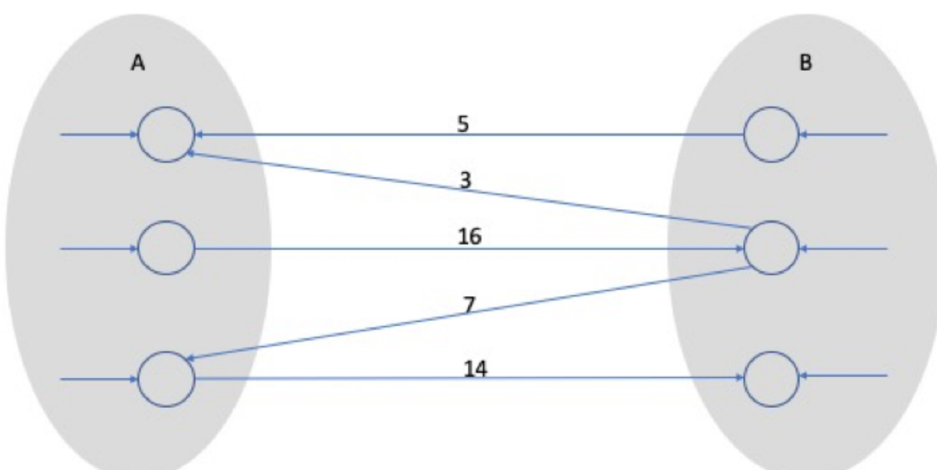

Figure 1: Question 1

- 15

- 45
- 30
- 10

1.1 Rationale

The flow-value lemma states that for any cut A, B where $s \in A$ and $t \in B$, the total flow of the network is given be the sum of flow exiting A minus the sum of flow entering A. Following the diagram, we have 30-15=15.

2 Problem 2

Node A and its adjoining edges have been extracted from a network wherein f_i and c_i represent the flow and capacity of each edge i respectively. Use the following diagram to determine the value of f_x given that node A has been extracted from a valid network.

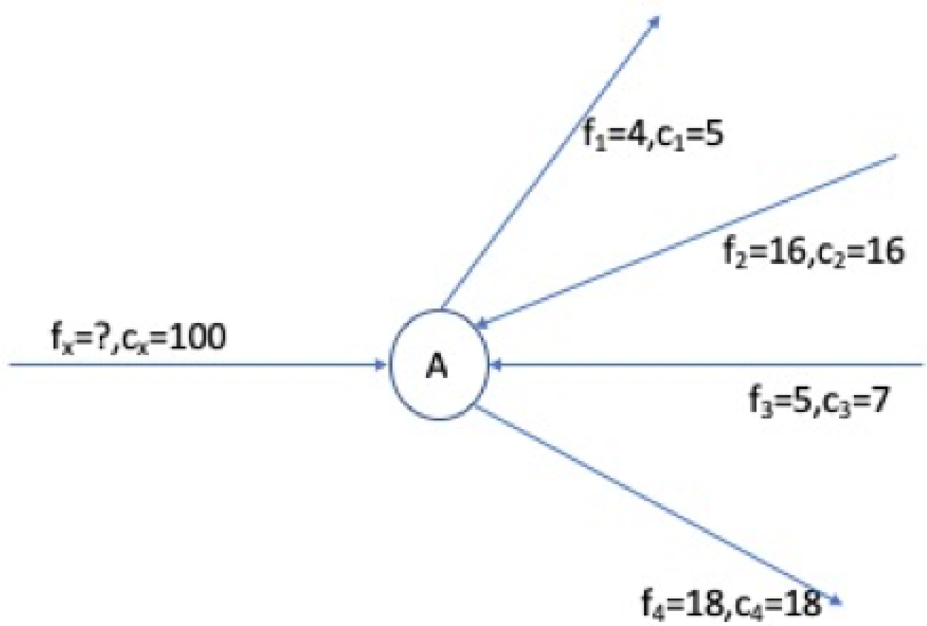

Figure 2: Question 2

- 1
- 21
- 22
- 0

2.1 Rationale

The flow conservation property states the sum of flow entering a node must equal the sum of flow exiting a node. Thus, the only way to satisfy f_x is with value 1 since the sum of flow exiting A is 22 and the sum of flow entering is $21{+}f_x$.

3 Problem 3

Suppose we are given a network partitioned into sets A and B such that source node $s \in A$ and sink node $t \in B$. This partition creates a cut in the network. Let the values along each edge represent the capacity of that edge. Determine the capacity of this cut.

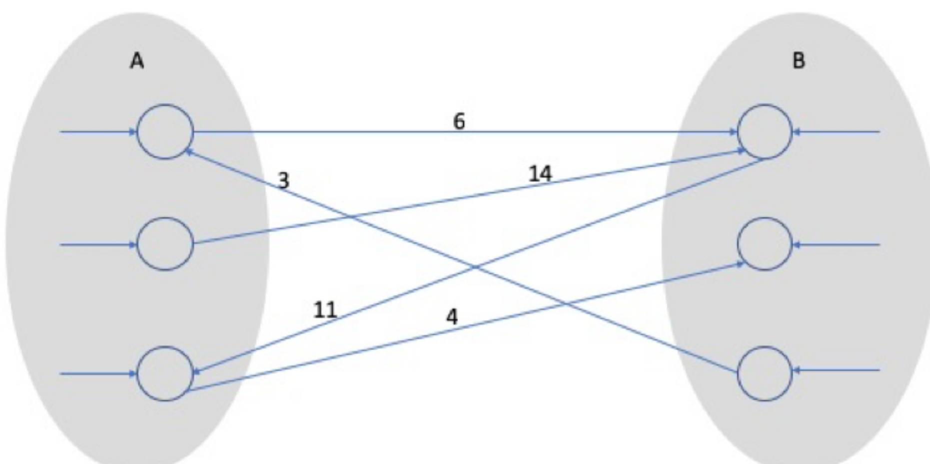

Figure 3: Question 3

- 24
- 14

- 27
- 38

3.1 Rationale

The capacity of an A, B cut where $s \in A$ and $t \in B$ is given by the sum of capacities of edges exiting A, which in this case is 6+14+4=24.

4 Problem 4

Given a network with source node s, sink node t, and the following general topology, find an expression for the number of $s - t$ cuts that exist in terms of the number of intermediary nodes i. Note that an intermediary node is any node that is not s or t.

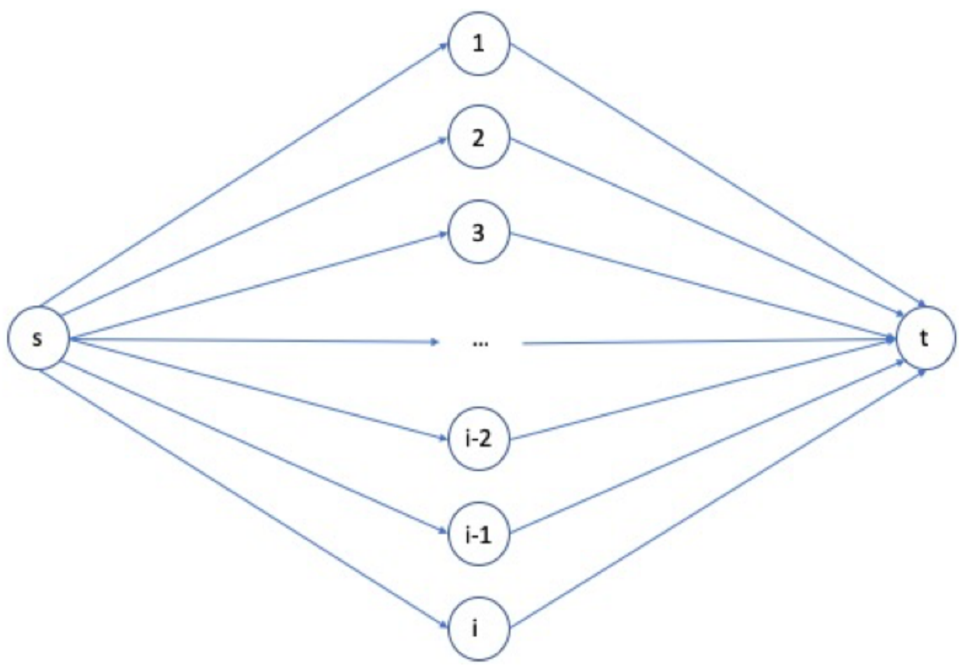

Figure 4: Question 4

- 2^i
- $(i)!$

- i^2
- i^i

4.1 Rationale

An s-t cut divides the set of vertices into sets A and B. Set A can be any possible subset of vertices 1 through i, and s. Thus, the number of cuts is equal to the cardinality of the power set of $\{1, 2, 3, \cdots, i\}$ where if A={s} then this represents the empty subset (recall the empty set is an element of every power set). The cardinality of a power set is exponential on the number of elements in the original set.

5 Problem 5

You are given the following flow network with source node s and sink node t as it stands in the middle of an execution of the Ford-Fulkerson algorithm after two paths have been augmented. You are given the corresponding residual graph at the same time step as well. Recall that we are trying to calculate the maximum flow value for this network. Suppose the next path the algorithm chooses to augment is the chain of nodes $P = S - A - D - B - C - E - T$. Give the updated flow values for each edge present in P after the augmentation is completed. Note that flow values are given in blue and capacity values are given in red. Also, recall the function $f((u, v))$ denotes the flow of edge (u, v).

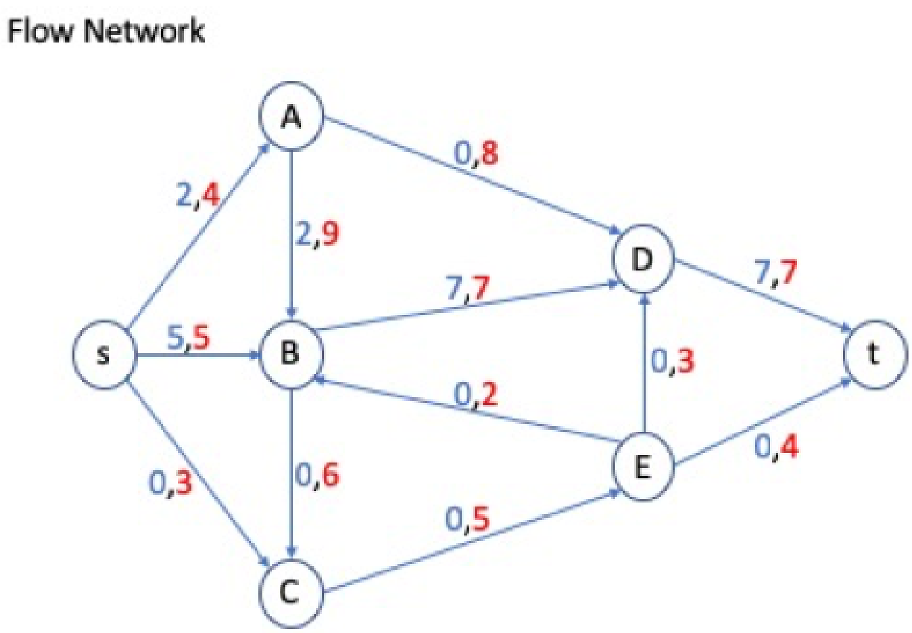

Figure 5: Question 5-1

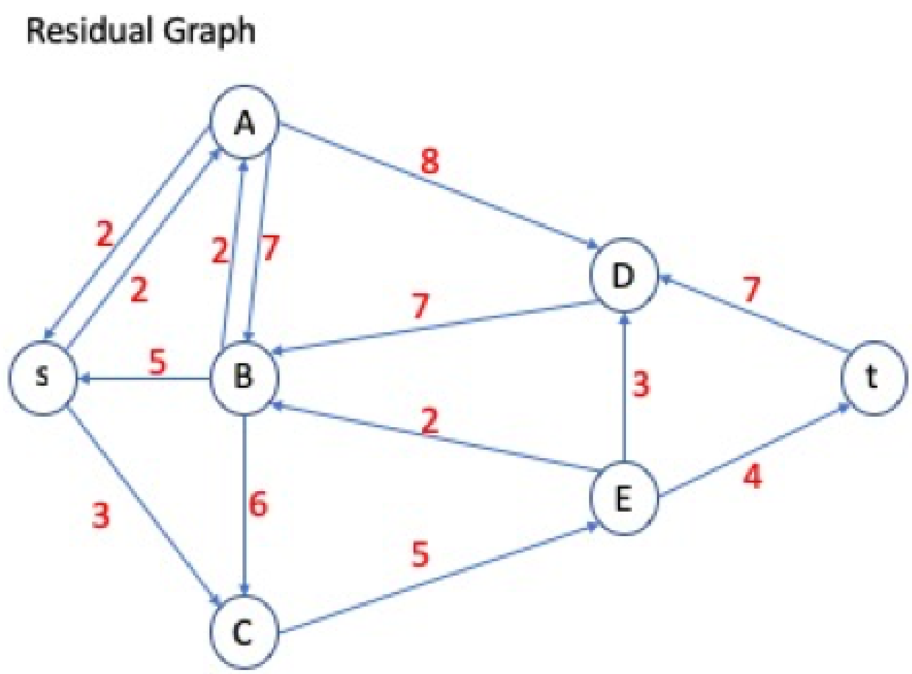

Figure 6: Question 5-2

- $f((S,A)) = 4, f((A,D)) = 2, f((B,D)) = 5, f((B,C)) = 2, f((C,E)) = 2, f((E,T)) = 2$
- $f((S,A)) = 4, f((A,D)) = 2, f((B,D)) = 9, f((B,C)) = 2, f((C,E)) = 2, f((E,T)) = 2$
- $f((S,A)) = 4, f((A,D)) = 10, f((B,D)) = 5, f((B,C)) = 8, f((C,E)) = 7, f((E,T)) = 6$
- $f((S,A)) = 4, f((A,D)) = 10, f((B,D)) = 9, f((B,C)) = 8, f((C,E)) = 7, f((E,T)) = 6$

5.1 Rationale

We follow the path s-A-D-B-C-E-t in the residual graph. For each edge of the path we check to see if the directed edge exists in the flow network. If so, augment flow along this edge in the flow network by the bottle-neck edge capacity, which is 2. Otherwise, decrement flow along the edge by 2 in the flow network.

6 Problem 6

You are given the following flow network with source node s and sink node t as it stands in the middle of an execution of the Ford-Fulkerson algorithm after four paths have been augmented. You are given the corresponding residual graph at the same time step as well. Recall that we are trying to calculate the maximum flow value for this network. Suppose the next path the algorithm chooses to augment is the chain of nodes $P = S - B - E - A - D - T$. Give the updated flow values for each edge present in P after the augmentation is completed. Note that edge weights in the flow network are given as u, v where u is the flow value assigned to that edge and v is the capacity of the edge. The edge weights of the residual graph represent capacities only. Also, recall the function $f((x, y))$ denotes the flow of edge (x, y).

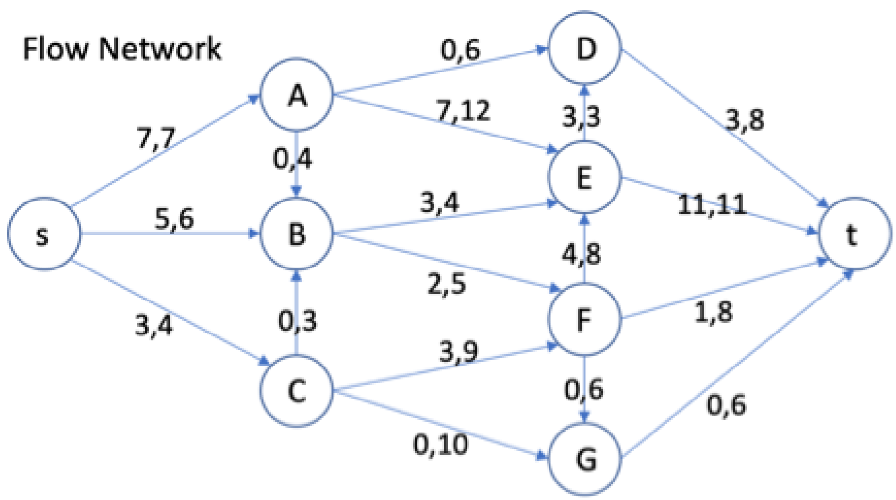

Figure 7: Question 6-1

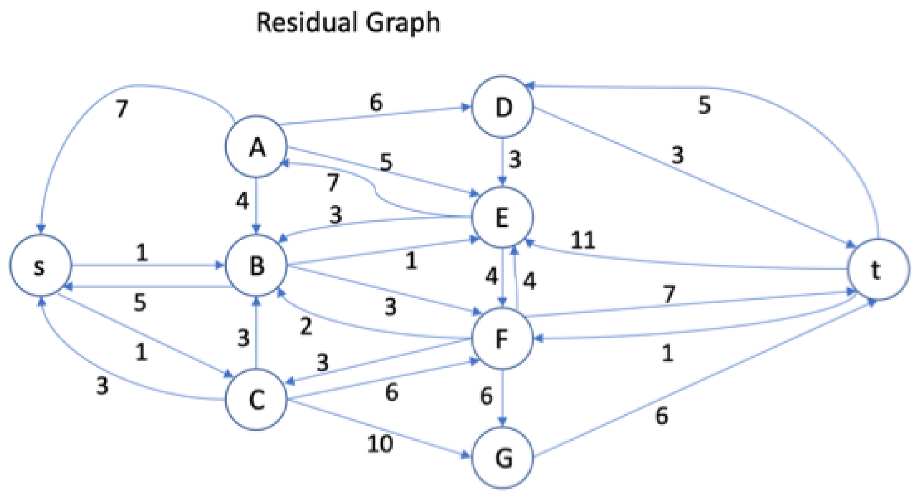

Figure 8: Question 6-2

- $f((S,B)) = 6, f((B,E)) = 4, f((A,E)) = 6, f((A,D)) = 1, f((D,T)) = 4$
- $f((S,B)) = 6, f((B,E)) = 4, f((A,E)) = 8, f((A,D)) = 1, f((D,T)) = 4$
- $f((S,B)) = 6, f((B,E)) = 2, f((A,E)) = 6, f((A,D)) = 1, f((D,T)) = 4$
- $f((S,B)) = 6, f((B,E)) = 2, f((A,E)) = 8, f((A,D)) = 1, f((D,T)) = 4$

6.1 Rationale

We follow the path s-B-E-A-D-t in the residual graph. For each edge of the path we check to see if the directed edge exists in the flow network. If so, augment flow along this edge in the flow network by the bottle-neck edge capacity, which is 1. Otherwise, decrement flow along the edge by 1 in the flow network.

7 Problem 7

Answer true or false to the following statement and explain: The Ford-Fulkerson algorithm runs in polynomial time.

- **False is only true if the capacities are some polynomial function of the number of nodes in the network.**

- False Because the capacity is represented with a logarithmic number of bits, FF always runs in exponential time.

- True The time complexity is the number of nodes in the network times the number of edges times the maximum capacity, which is a cubic function.

- True Because capacities are always constant, Ford-Fulkerson runs in polynomial time.

7.1 Rationale

The Ford-Fulkerson algorithm runs in pseudo-polynimal time. C represents the capacity function. Since we assume the network is connected, every vertex is incident to at least one edge. We need at least one bit to represent the value of each edge capcity, thus we need at least n bits to represent C. It only requires $\log C$ bits to represent C, so we have $n \leq \log C$ or $C \geq 2^n$. If we can guarantee C is a constant, or polynomial function of n, then we can say it runs in poly-time.

8 Problem 8

The following directed graph is a network with source node s and sink node t. The edge weights represent capacities along each respective edge. Use the max flow-min cut theorem to determine the maximum possible flow of this network. Alternatively, you may apply the Ford-Fulkerson algorithm.

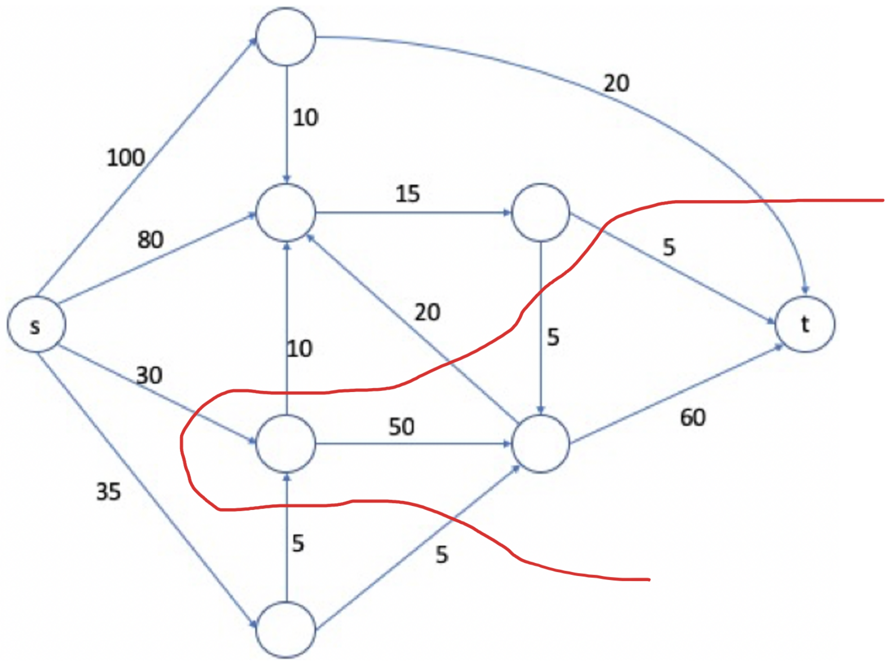

Figure 9: Question 8

- 70
- 90
- 100
- 85

8.1 Rationale

The s, t-cut with capacity 70 exists and is given by the red line above. Recall that the capacity of a cut A, B is the sum of all edge capacities exiting A. Since flow flows from s to t, $s \in A$, and the capacity of the minimum cut is 70.

9 Problem 9

The most efficient algorithm that currently exists to solve the max-flow problem is the Edmonds-Karp algorithm. This algorithm selects augmenting paths with the fewest number of edges. By which mechanism does Edmonds-Karp achieve this?

- Breadth-First Search
- Depth-First Search
- Both of Breadth-first search or Depth-first search work
- Neither breadth-first search nor depth-first search work

9.1 Rationale

According to the slides, the answer is breadth-first search.

10 Problem 10

Which factor of the Capacity scaling algorithm ensures the number of outer while-loop iterations is logarithmic on the capacity?

- Δ is reduced by half after each iteration of the loop
- Δ is initialized to the power of two nearest to the capacity
- The Δ-residual graph at each iteration is smaller than the corresponding flow network.
- There is a logarithmic number of bits required to represent the capacity

10.1 Rationale

Let's analyze the following code:

```
int c = 0;
for(int i = 32; i ≥ 1; i=i/2){
c++;
}
printf(c);
```

Since the value of i decreases by half each iteration, the loop terminates after log(32)=5 iterations, which is also the value of variable c. Only the highlighted answer has anything directly to do with capacity scaling's runtime.

11 Problem 11

Determine which of the following statements is true regarding the bipartite matching application of the network flows problem.

- **The capacity of the minimum cut is equal to the maximum cardinality of the bipartite matching**
- The number of edges between node sets L and R must be greater than the number of edges from source node s to each node in L.
- The algorithm execution will fail to find a maximum cardinality bipartite matching if the capacity of each edge traversing node sets L and R is one.

11.1 Rationale

Statement 1 is correct and was proven in lecture. Regarding statement 2, this is not necessarily true. The number of edges between L and R can also be equal to the number of edges from s to L. Regarding statement 3, the edges connecting L to R can be any value of at least 1.

12 Problem 12

Given the following flow network, use the max-flow/min-cut theorem to determine the number of edge disjoint paths in the network. Note that the edge values represent capacities.

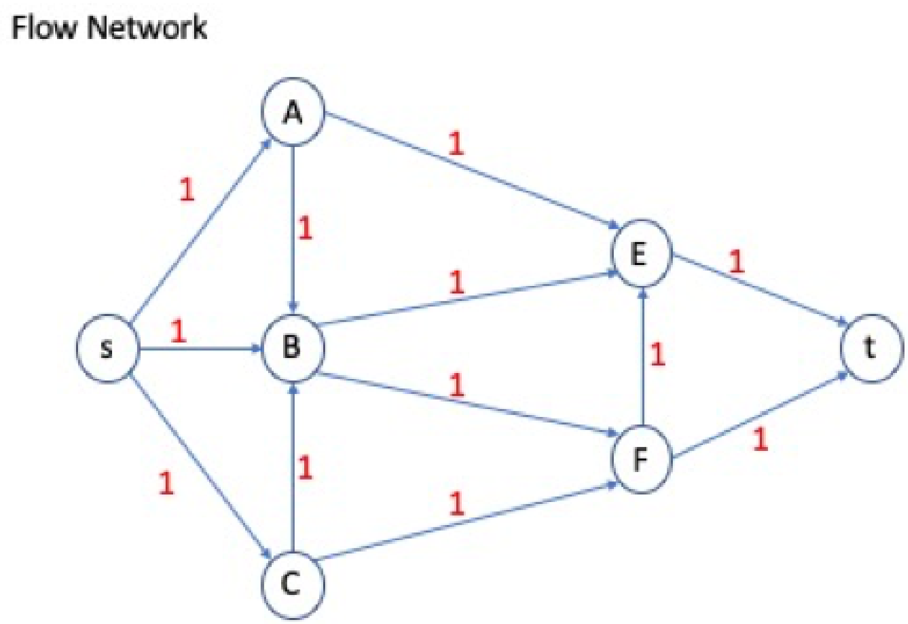

Figure 10: Question 12

- 2
- 3
- 4
- 5

12.1 Rationale

It is easy to see the max-flow/min-cut of this network has value 2. It was proven in lecture that this value is equal to the number of edge disjoint s-t paths.

13 Problem 13

Given a flow network $G = (V, E)$ with flow assignment f, edge capacities c, and a corresponding residual network $G' = (V', E')$, which of the following describes the relationship between E and E'?

- The capacity of an edge $e' \in E'$ is equal to $c(e) - f(e)$ for a corresponding $e \in E$.
- The capacity of an edge $e \in E$ is equal to $c(e') - f(e')$ for a corresponding $e' \in E'$.
- The capacity of an edge $e' \in E'$ is equal to $c(e) - f(e')$ for a corresponding $e \in E$.
- The capacity of an edge $e \in E$ is equal to $c(e) - f(e')$ for a corresponding $e' \in E'$.

13.1 Rationale

The first response is correct. Recall that $e \in Ee' \in E'$ but the converse does not necessarily hold. Then the capacity of an edge $e' \in E'$ represents the residual capacity of $e \in E$ under some flow assignment. The residual capacity is simply the remaining capacity of e under flow assignment f, or simply $c(e) - f(e)$ (which may be zero).

14 Problem 14

Identify which of the following statements are true regarding any flow network $G = (V, E)$. **Select all that apply**.

- G may have multiple minimum cuts.
- G necessarily has at least one cut.
- G may not have a maximum flow.
- The Ford-Fulkerson algorithm is guaranteed to run in polynomial time in the size of G when given G as an input.

14.1 Rationale

Statement one is clearly true. Take an example of a network that is a directed line from source s to sink t, and let all capacities be 1. Then any edge of such a network corresponds to a minimum cut. Statement two is true since a flow network is defined to have at least two nodes (a source and a sink). In the case of exactly two nodes, the minimum cut is simply defined by the single directed end from source s to sink t. Statement three is false since the zero-flow (i.e. assign zero flow to all edges) is a valid flow assignment since it satisfies the flow conservation constraints and does not violate edge capacities (as they are necessarily non-negative). Regarding statement four, we have seen in lecture that the standard Ford-Fulkerson algorithm is a pseudo-polynomial time algorithm.

15 Problem 15

Given a flow network $G = (V, E)$, edge capacities c, and a maximal flow assignment f, which of the following scenarios are possible? **Select all that apply**.

- $f(e) = c(e), \forall e \in E$
- For some cut C of G, the flow along an edge of C is strictly less than the capacity of that edge.
- For a minimum cut $S, \bar{S}$ of G, the flow along an edge exiting S is strictly less than the capacity of that edge.
- For a minimum cut $S, \bar{S}$ of G, the flow along an edge entering S is equal to the capacity of that edge.

15.1 Rationale

Statement one is correct. Again, consider the flow network that is a directed line from source s to sink t where all edge capacities are 1. There is only one max-flow assignment, and it saturates every edge. Statement two is also correct. Any non-minimum cut must have some unsaturated edge by the max-flow/min-cut theorem. From this also follows that statement three is false. Regarding statement four, recall that the flow of a network can be computed from any cut by summing the flow leaving S and subtracting the flow entering S. But $S, \bar{S}$ is a minimum cut and f is a maximum flow assignment, and so by the max-flow/min-cut theorom, the capacity of $S, \bar{S}$ is equal to the value of f. If there were any edges carrying flow into S, then $S, \bar{S}$ would not be a minimum cut since the capacity of a cut is defined as the sum of capacities of outgoing edges of S only. Thus, we have a contradiction.