Section 7: Max Flow Min Cut and Applications

The goal for this week's section is to use max-flow and min-cut to model various problems. There are a few common tricks that allow us to cleverly solve problems that at first glance look like they are *not quite* the same as the "standard" problems. You might figure out the trick the first time, or you might not. But either way you should remember these tricks! If you've seen it once, it's much easier to use later, and these tricks are common in modeling problems with flows and cuts.

1. You're not a dummy...

You have three overfull reservoirs and two underfull reservoirs. You want to (as quickly as possible) move a total of 10,000 gallons of water from the overfull reservoirs to the underfull ones. You do not care how much comes from each of the three individual reservoirs (as long as the total is 10,000 gallons) nor how much arrives at each of the underfull ones (again, as long as the total is 10,000 gallons). You have a map of (one-way) pipes connecting the reservoirs (in the form of a directed graph); each pipe has a maximum capacity in gallons per minute. You wish to find the way to route the water and the amount of time that will be required.

1.1. Read The Problem Carefully

Answer the usual quick-check questions:

- Are there any technical terms in the problem you don't know? Are there any words that look like normal words, but are actually technical terms?
- What is the input type?
- What is the output type?

1.2. Make a basic model

This sounds like a flow problem. From what you know so far, what would the flow model be? What parts of the problem have you represented successfully? What is still missing?

1.3. Brainstorm: How can you fix the missing piece?

Find a clever trick to represent the missing piece. The goal here is to do a reduction. By the end of this step, you should have a "standard" flow problem.

1.4. Correctness and running time

Explain why your algorithm is correct. For flow problems, the proof is usually just explaining how you've represented each part of the problem, and relying on the correctness of the flow algorithm.

2. Split Personality

You have been given a map of the water cleaning system for the city of Seattle. Water enters from a marked vertex, and flows through pipes (directed edges with specified capacities), through processing facilities, and back out to nature (marked as a specified sink vertex). The processing facilities are vertices in your graph. As the facilities process the water, they **also** have maximum capacities, which may be less than the total capacity entering or leaving the vertex. You wish to find the amount of water that can flow through this network while respecting both the facility and pipe capacities.

2.1. Read The Problem Carefully

Answer the usual quick-check questions:

- Are there any technical terms in the problem you don't know? Are there any words that look like normal words, but are actually technical terms?
- What is the input type?
- What is the output type?

2.2. Make a basic model

This sounds like a flow problem. From what you know so far, what would the flow model be? What parts of the problem have you represented successfully? What is still missing?

2.3. Brainstorm: How can you fix the missing piece?

Find a clever trick to represent the missing piece. The goal here is to do a reduction. By the end of this step, you should have a "standard" flow problem.

2.4. Correctness and running time

Explain why your algorithm is correct. For flow problems, the proof is usually just explaining how you've represented each part of the problem, and relying on the correctness of the flow algorithm.