Problem Set 6 - Network Flow

Problem B

Magic Potion was the first question that I got for Network flow. I reckon it's one of the most intuitive applications of Network Flow in that all we're trying to do is set up the flow network in a way that the final output max flow gives us the optimum/max allocation, not dealing with weird min-cut bottleneck stuff as in Question A.

If we are to ignore the magic potion for now, the question would be a simple bipartite matching problem, with the soldiers on the left being linked to who they can kill on the right. Since the soldiers can only kill one monster each, they would have an incoming node from the source with weight = 1, and since each monster can only be killed once, they would have an edge going into the sink with weight one. And when we run flow, we see how many monsters can be killed.

The next step was to figure out how to implement the magic potion into the flow network. Two properties to note about the potion - they can make a soldier kill another monster, and they can only be used once per soldier. Because of this, we can think about the potions as creating a duplicate/shadow clone soldier - with a cap on how many distinct soldiers can be duplicated. The way to achieve this, is to just create another potion node with a limit (edge from sink to it) of k. This node would be linked with edges of cap 1 to another set of duplicate soldier nodes, which flow to the monsters from our original graph.

Finally, with this, we can run flow again in order to get the number of monsters that can be killed with the potions included. I don't think I had any other initial incorrect approaches about this question - if I did, they were so short lived that they've since been forgotten.