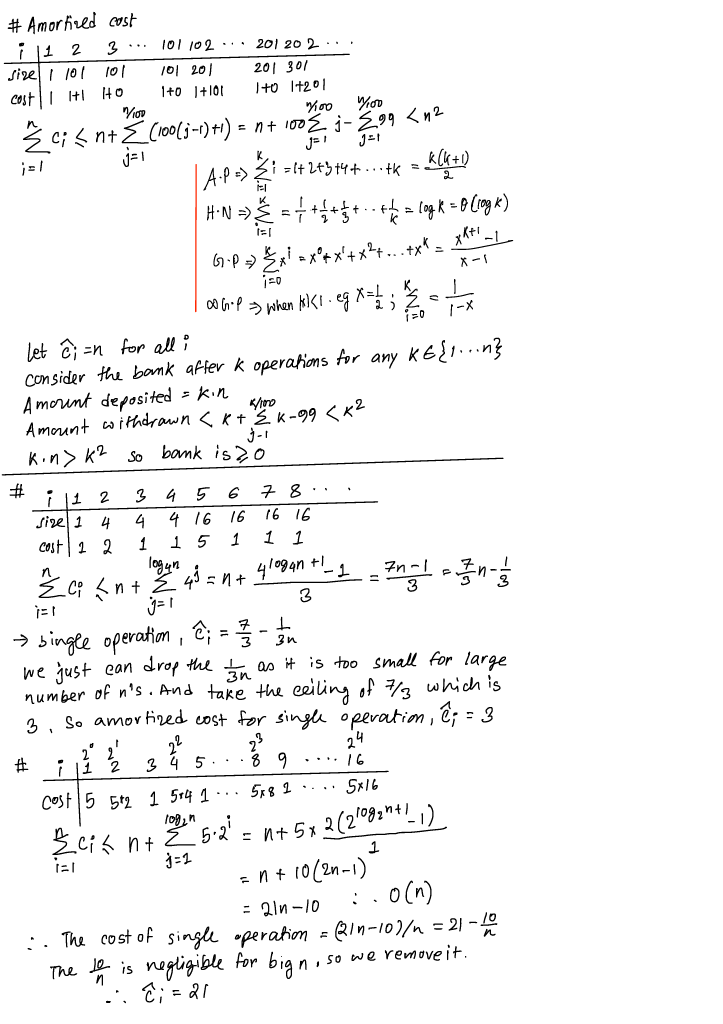
A paper with text and black text

Description automatically generated with medium confidenceA paper with writing on it

Description automatically generatedA paper with text on it

Description automatically generatedA paper with writing on it

Description automatically generated

For this problem set I will use the Kruskal algorithm. It will automatically sort the edges in non-decreasing order, so when we find the minimum spanning cost, we just have to save the last cost to go from u to v and then use free edge coupon in it. We can achieve it by subtracting the last (maximum edge) and reducing it from the total cost.

First sort all the activities according to its end point. Run a while loop through all the activities until none left. Inside the while loop, run a for loop where it checks if any activities that doesn’t overlap. If not overlaps, it removes the activities from the activities array and then increase the resource count by 1.