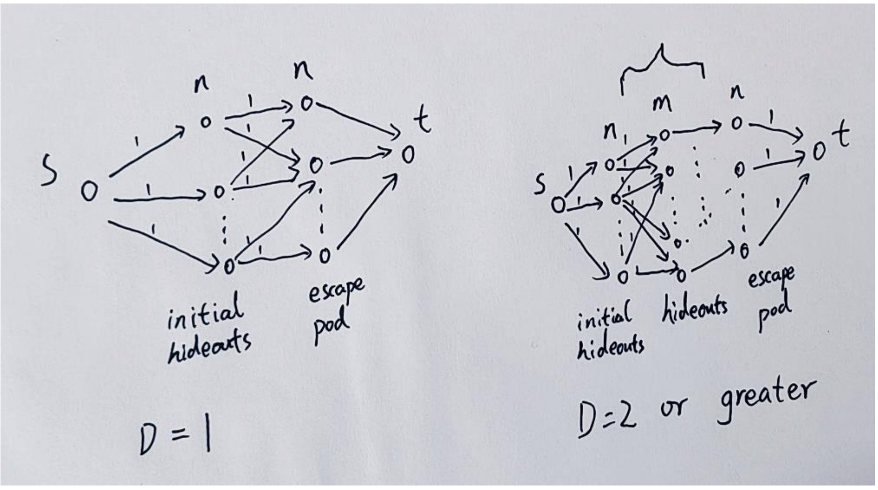
Qiyao Zhou

Z5379852

Question 4

4.1 We first take care of the case where D = 1. We begin by constructing a flow network using the input data, label the initial hideouts h [1], h [2], · · ·, h[n], and the hideouts vertex which has emergency escape pod p [1], p [2], · · ·, p[n]. The network is showed in the diagram below:

- source s and sink t,
- the combinations of vertices from left to right represent initial hideouts and hideouts vertex which has emergency escape pod,
- connect s to each initial hideouts vertex with capacity equal to 1,
- connect h[i] and p[j] which have tunnels between(T[i][j] =True) with capacity equal to 1(Note that we also need to add an edge of capacity 1 when h and p represent the same node),
- connect each hideouts vertex which has emergency escape pod to t with capacity equal to 1,



From this flow network construction, we run Ford-Fulkerson to find the maximum flow. If the maximum flow is less than n, then we output “no solution”. Otherwise, it is possible for all spies to escape successfully.

This can be extended to the case where D is a value of 2 or even greater. We can add layers which consists of all hideouts vertex between h and p and connect each other which have tunnels between.

Then, we can run Ford-Fulkerson to find the maximum flow. If the maximum flow is less than n, then we output “no solution”. Otherwise, it is possible for all spies to escape successfully.

The time complexity is O(|V||F|) where V ∈ n and F $\in m^2D$, so the algorithm runs in O (nm^2D).

4.2 According to the algorithm in 4.1, we can use a binary search to find the minimum number of days by decreasing the number of intermediate levels one at a time, starting with D, and run Ford-Fulkerson after each search to find the maximum flow until the maximum stream is found as the minimum value of n. At this point the number found is the minimum number of days.

The time complexity is O (nm^2D)logD = O (nm^2DlogD).