Semi-streaming: Allow $O(n \text{ poly} \log n)$ space, multiple passes.

① Number of CCs:

- Goal: To determine the # of CCs for a given graph $G=(V,E)$
- High level: Construct a spanning forest F of G as follows:
  - Join two nodes u,v which are not already connected (otherwise, we would create a cycle.)

Algorithm 1: Construct F from G

1. $F \leftarrow \{\}$
2. for each edge $\{u,v\} \in E$:
3.   if u and v are not connected in F:
4. $F \leftarrow F \cup \{\{u,v\}\}$
5. return F

Note:

- F has as most $n-1$ edges
- With Union-find data structure, we can count the # of CCs using $O(n \log n)$ space.

2) k-Edge Connectivity:

- Def: A graph is said to be "k-edge connectd" if after removing any $k-1$ edges, it remains connected.
- High level:
  - Start off by maintaining k distinct forests of G: $F_1, F_2, \ldots, F_k$.
  - Then, we stream one edge at a time from G:
    - For each edge $\{u,v\} \in E$, we find the smallest F_i in which u and v are not connected. Then, add $\{u,v\}$ to such F_i
- Analysis:
  - Formally, we compute $H = F_1 \cup F_2 \cup \ldots \cup F_k$. So, H is a subgraph of G, and as a result $E(H) \subseteq E(G)$. We claim that G is k-connected iff and only if H is k-connected.

- Consider any cut $S \subset V$. Let $\delta_G(S)$ denote the set of edges crossing this cut in G. We consider $\delta_H(S)$.
- Since H is a subgraph of G, $|\delta_G(S)| \geq |\delta_H(S)|$
- Suppose $\exists \{u,v\} \in \delta_G(S)$ s.t. $\{u,v\} \notin \delta_H(S)$. This edge appeared in the stream but we ignored it as every forest $F_1, F_2, \dots, F_k$ already had a connection between u, v
- Therefore, in each F_i, there is one edge crossing the cut S. Since all the k forests are disjoint, $|\delta_H(S)| \geq k$.
- On the other hand, if there is no such edge, then $|\delta_H(S)| = |\delta_G(S)|$.
- Combining the two arguments, we get

$$|\delta_H(S)| \geq \min(|\delta_G(S)|, k)$$

- If G is k-connected, then $\forall S \subset V$, $\delta_G(S) \geq k$, and so $\delta_H(S) \geq k$. If G is not k-connected, then $\exists S \subset V$, $\delta_H(S) < k$, and so $\delta_G(S) \leq k$.

<u>Note:</u> - The algorithm takes $k+1$ passes to construct $F_1, \dots, F_k$. Each pass requires $O(n \log n)$ space. The total space used by the algorithm is $O(k \cdot n \cdot \log n)$