Problem 1 — SCC and Reversal

1. Algorithm to compute the reversal rev(G)

For a directed graph $G = (V, E)$:

Create a new graph with the same vertices.

For each edge $(u \to v) \in E$, add edge $(v \to u)$ to the new graph.

This processes each vertex and each edge once, so the time complexity is **O(V + E)**.

2. The strong component graph scc(G) is acyclic

In scc(G), each node represents a strongly connected component of G.

Assume scc(G) has a cycle.
Then all components in the cycle can reach each other, which means they form one larger strongly connected component. This contradicts the definition of SCCs.

Therefore, **scc(G) is acyclic**.

3. Proof that scc(rev(G)) = rev(scc(G))

- Reversing G reverses all paths.
- Strong connectivity depends on mutual reachability, which is preserved under reversal.
- Therefore, SCCs remain the same, but edges between components are reversed.

Hence, reversing first and then contracting SCCs gives the same result as contracting first and then reversing:

$$scc(rev(G)) = rev(scc(G))$$

4. Reachability and SCC graph

Let $S(v)$ be the SCC containing vertex v.

- If u can reach v in G, then there is a path between their SCCs in scc(G).
- If $S(u)$ can reach $S(v)$ in scc(G), then there exists a path from any vertex in $S(u)$ to any vertex in $S(v)$ in G.

Thus:

$$u \text{ reaches } v \text{ in } G \iff S(u) \text{ reaches } S(v) \text{ in } scc(G)$$

Problem 2 — Euler Tour

1. Condition for existence of Euler tour

A strongly connected directed graph has an Euler tour **if and only if**:

$$\text{in-degree}(v) = \text{out-degree}(v) \forall v \in V$$

If degrees differ, some edges cannot be entered and exited equally.

2. O(E)-time algorithm to find an Euler tour

Start from any vertex.

Follow unused outgoing edges until returning to the start (forms a cycle).

If unused edges remain, start a new cycle from a vertex on the existing tour.

Merge cycles.

Continue until all edges are used.

This is **Hierholzer's algorithm**, which runs in **O(E)** time.

Problem 3 — Topological Sort

Given graph:

$$\{A \to B, A \to C, B \to C, B \to D, C \to E, D \to E, D \to F, G \to F, G \to E\}$$

Topological order starting from A

One valid ordering:

A, B, D, C, E, F, G

Another valid ordering (starting from G)

G, A, B, D, C, F, E

Another possible ordering

A, G, B, C, D, E, F

Multiple orderings are valid as long as all edge directions are respected.