

Q1

We would need  $O(n^2)$  Space to represent all nodes in the adjacency matrix. We will create a  $n \times n$  matrix. That is enough to represent all vertexes and the edges between them.

Q2

If we have Rows  $\neq$  Columns, then it would make no sense to make the adjacency matrix. We won't be able to represent complete relationships between vertexes. e.g: If we have a relation with a last vertex that is added to graph, we can't represent its relationship with others and vice versa.

Q3

Topological Sort can't be applied to undirected graphs. It relies on the directionality of edges to determine the linear ordering, and since undirected graphs don't have directed edges, it's not possible to satisfy the required condition that for every directed edge  $(u, v)$ , vertex  $u$  must come before vertex  $v$  in the ordering.

Q4

It cannot be applied on cycles. In topological sort, all vertices must come before their adjacent vertices. The cycle introduced the circular dependency, which makes it difficult to apply topological sort.

Q5

We have different algorithms to detect a cycle. We can apply DFS and traverse the graph. If we encounter any node that is already visited, during traversing then, we can say that our graph has cycle.