

**1. Given an adjacency list, how can you convert it to an adjacency matrix?**

We can convert an adjacency list to a Adjacency Matrix by using a 2D array of size  $V \times V$  where  $V$  is the number of vertices in a graph.

We have to let the 2D array be `adj[][]`. A slot `adj[i][j] = 1` indicates that there is an edge from vertex  $i$  to vertex  $j$ .

- Initialize a matrix with **0s**.
- Iterate over the vertices in the adjacency list
- For every **jth** vertex in the adjacency list, traverse its edges.
- For each vertex **i** with which the **jth** vertex has an edge, set `mat[i][j] = 1`.

**2. Given an adjacency matrix, how can you convert it to an adjacency list?**

To convert an adjacency matrix to the adjacency list:

- Create an array of lists and traverse the adjacency matrix.
- Look for any cell **(i, j)** in the matrix that is not equal to zero “`mat[i][j] != 0`” (it means there is an edge from **i** to **j**)
- Insert **j** in the list at **i-th** position in the array of lists.

**3. Given a directed graph, how can you reverse the direction of each edge?**

To change the direction of the edges on a directed graph we:

- Traverse the adjacency list
- As we find a vertex **v** of vertex **u** (meaning there is an edge from **u** to **v**)
- Add a edge from **v** to **u**
- Add **u** to adjacency list of vertex **v**