Q1

Big-Oh Space complexity of adjacency lists will be **O(V+E).** Here, V is the total vertices and E are the edges. Each node in the list will store all nodes of the graph. And corresponding to them, the associated edges will be stored in the list.

Q2

Big-Oh Space complexity of adjacency matrices will be **O(V^2).** This is because V x V Matrix is generated. Whether the edges between vertices exist or not, the matrix will be created, and space will be occupied.

Q3

The time complexity of using a DFS for traversing a graph is O(V+E). It traverses all vertices and edges of the graph, that's why this complexity is obtained. The representation of the graph can make a difference in constant factors. Adjacency list is typically more sparse representation of graphs. It allows quicker access to each vertex.

Q4

The time complexity of using a BFS for traversing a graph is O(V+E). It traverses all vertices and edges of the graph, that's why this complexity is obtained. The representation of the graph does affect the time complexity in practice. However, using an adjacency list is generally more efficient especially in the case of sparse graphs as it directly accesses the neighbors without checking every possible edge, which is the case in an adjacency matrix.