1

The big-Oh space complexity of an adjacency list is O(V+E). V is the number of vertices and E is the number of edges. It needs O(V) memory to store all vertices and O(E) to store all edges. Therefore, the space complexity is O(V+E).

2

The big-Oh space complexity of an adjacency matrix is O(V^2). It needs V rows and V columns to store 1 or 0 for each pair of nodes.

3

The representation of the graph can affect the time complexity. With an adjacency list, the big-Oh time complexity for searching an entire graph using DFS is O(V+E). It needs O(1) time to visit a node and O(e) time to visit all edges starting from this node. If there are V vertices, the sum of the edges of each node e is the number of edges of the graph E. Thus, the time complexity is O(V+E). With an adjacency matrix, the time complexity is O(V^2) because all cells in the V columns and the V rows will be visited once.

4

Similar to DFS, the representation of the graph can affect the time complexity. With an adjacency list, the big-Oh time complexity for searching an entire graph using BFS is O(V+E). With an adjacency matrix, the time complexity is O(V^2).