Mod 6 Prompt

Q1.
Sparse matrices often can be seen in preprocessing categorical data for Machine learning using 1-hot encoding. Given a categorical column in tabular data, it can be encoding as a sparse matrix that each column represents a unique value from that category. If the row is matching that corresponding category, that intersect of the row and columns field will be 1.

Q2.
Since the sparse matrices have majority fields of the matrix as 0 value, in most cases the 0s doesn't represent specific scaler meanings. So for its implementation, it's important to consider if there is any way to save the cost (time & space) while storing and manipulating.
I believe it's crucial that the data structure needs to store non-zero field's values and indices effectively.

Q3.
Pros of the linked list implementation of a sparse matrix:
Less data is stored while representing the same structure. Searching is faster since sequential scanning of the linked list will skip the zero field in a sparse matrix. Also the size is not fixed so that the matrix can grow.

Cons:
No random access. The method for operating basic manipulations such as matrices addition, row & column swap, are more complicated than the array based implementation.

Q4.
If there's many relationships between multiple data and need to parse/traverse through in order of that relations, I will choose to use a graph. Specifically, needs to consider what are vertices, edges and sometimes weights of the edges.

Q5.
A linked list is a direct way to represent a graph, since while constructing we can use the left, right, center pointers to point another node/element to represent edges and vertices.
Another advantage will be dynamic size so the graph can grow. Cons of the linked list implementations are: Random access is not allowed, and to access a specific element in the graph, the program needs to traverse the graph.

Q6.

We can use array based implementation for the graph, if the graph follows specific characteristics such as binary tree. So the program can randomly access and manipulate any node in the tree by the characteristics and patterns of indices. However, the array is fixed size so that the graph can not expand with more elements.

Q7.

LinkedIn, or any social networking services utilize graphs to represent connections between each user. The graph information can provide networking(friendship) recommendations for the users.

Q8.

Since the graph has vertices that usually share the same type attributes, such as personal information for SNS use cases, it’s possible that the program recursively access these attributes for multiple vertices and manipulate the similar or same operations to these.
Also traversing the graph can be operated recursively.