Data Structures Reflection:

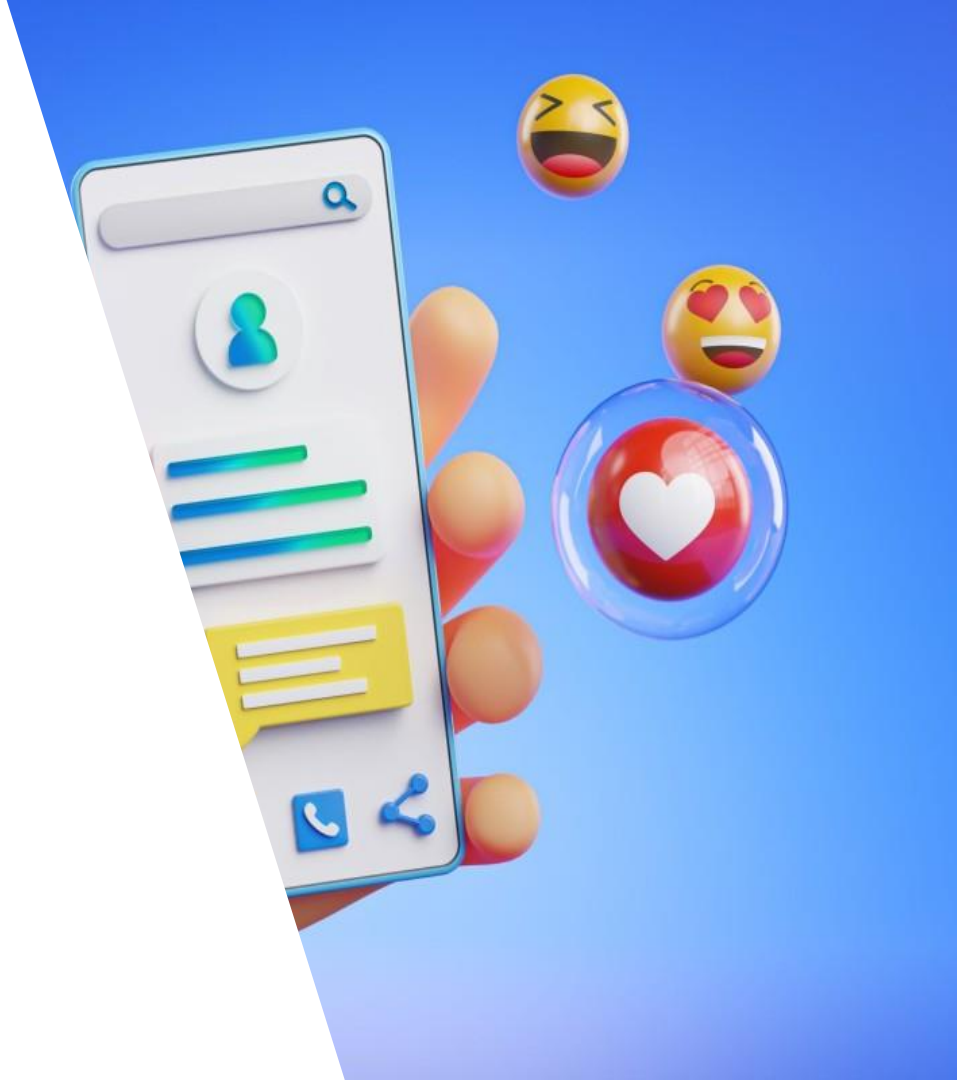
Social media platform

Unit 3 e-Portfolio Activity

by Andrius Busilas

Introduction

When considering an online system that we use daily, such as a social media platform, we can analyze how various data structures operate at the back-end to manage data efficiently. Here are some key data structures that might be involved:



Graphs

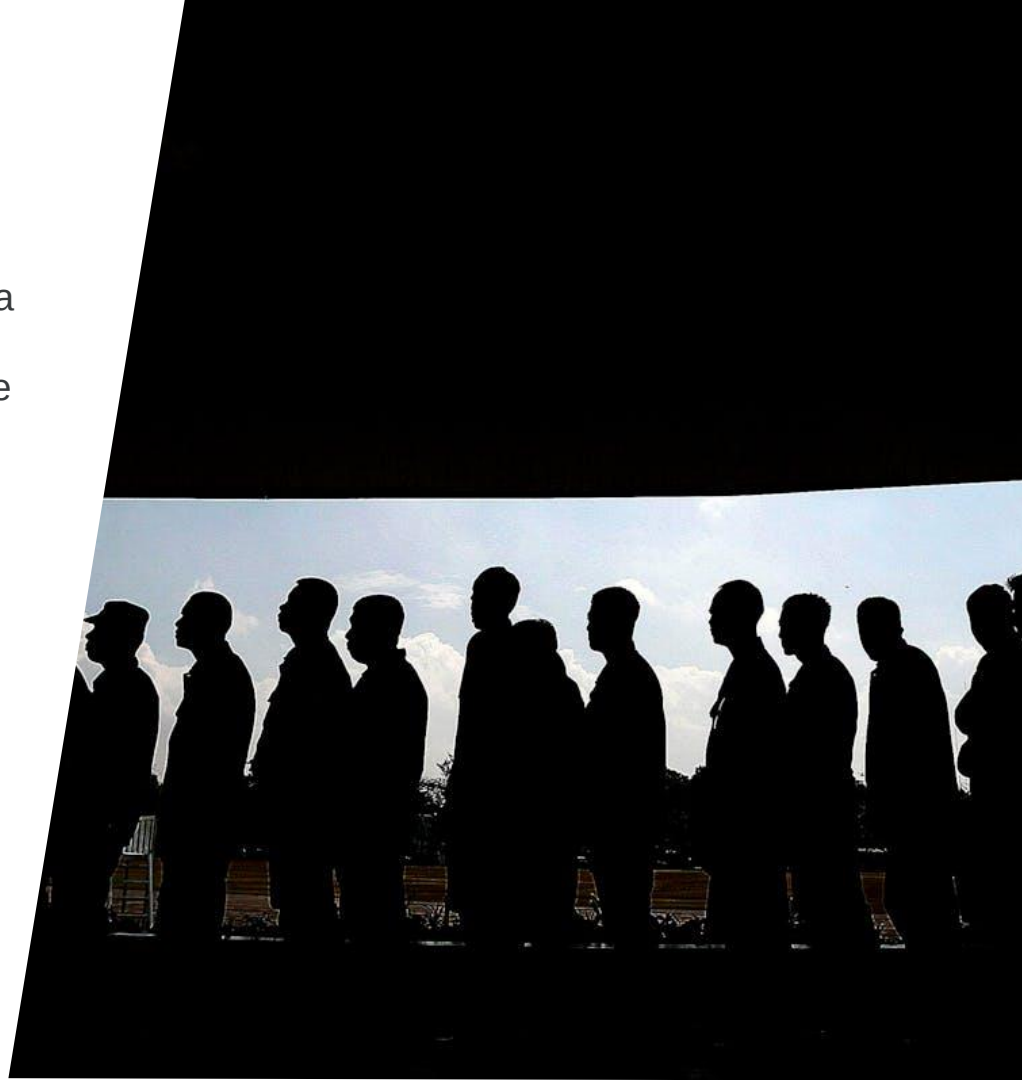
Social media platforms often use graph data structures to represent users and their connections (friends, followers). Each user can be a node, and the relationships (friendships, follows) can be edges. This allows for efficient traversal and querying of user connections, such as finding mutual friends or suggesting new connections.

Hash Tables

These are commonly used for quick lookups of user profiles, posts, or comments. For instance, when a user searches for a friend or a specific post, a hash table can provide O(1) average time complexity for retrieval, making the search process fast and efficient.

Queues

Queues can be used to manage notifications or messages. For example, when a user receives a new message or notification, it can be added to a queue, ensuring that they are processed in the order they were received. This is particularly useful for managing real-time updates.



Hash Tables

Data structures like binary search trees or B-trees can be used for organizing posts or comments. For instance, a binary search tree can help in efficiently storing and retrieving posts based on timestamps or popularity, allowing users to see the most relevant content quickly.

Arrays and Lists

These are fundamental for storing user-generated content, such as posts, comments, and likes. Dynamic arrays or linked lists can be used to manage the growing number of posts and comments, allowing for efficient insertion and deletion.

Stacks

Stacks might be used in features like "undo" actions or managing the history of user interactions. For example, if a user deletes a post, the action can be pushed onto a stack, allowing them to undo the deletion if needed.

Conclusions

By understanding how these data structures work together, we can appreciate the complexity and efficiency of the back-end operations that support the user experience in online systems. This knowledge will be valuable for our seminar discussion, as we can explore how different data structures impact performance, scalability, and user interaction.