**🚀 Day 15 of the Striver Sheet Challenge! 🚀**

Today, I delved into some exciting graph concepts and algorithms. Here’s a quick recap:

1. **📚 Graph Basics**:
   * **Definition**: A graph is a collection of nodes connected by edges, used to model various relationships.
2. **🔍 Types of Graphs**:
   * **Directed vs. Undirected** 🌐
   * **Weighted vs. Unweighted** ⚖️
   * **Cyclic vs. Acyclic** 🔄
3. **🗂️ Graph Representation**:
   * **Adjacency Matrix** 🧩
   * **Adjacency List** 📝
   * **Edge List** ✏️
4. **🔎 Graph Traversals**:
   * **DFS (Depth-First Search)**: Explores as far as possible along each branch before backtracking. 🚀
   * **BFS (Breadth-First Search)**: Explores all neighbors of a node before moving to the next level. 🌊
5. **🔄 Cycle Detection**:
   * **Undirected Graphs**:
     + DFS 🕵️
     + BFS (Bipartite Check) 🔄
     + Disjoint Set Union (DSU) 🔗
   * **Directed Graphs**:
     + DFS using Recursion Stack 🧩
     + BFS using Kahn's Algorithm 🔄
6. **📏 Shortest Path Algorithms**:
   * **Dijkstra’s Algorithm**: Shortest path for positive weights. 📍
   * **Bellman-Ford Algorithm**: Handles negative weights and detects negative weight cycles. ⚠️
   * **Floyd-Warshall Algorithm**: Shortest paths between all pairs of nodes. 🌐
7. **🌲 Minimum Spanning Trees (MST)**:
   * **Prim’s Algorithm**: Builds MST by adding the smallest edge connecting to a new node. 🌿
   * **Kruskal’s Algorithm**: Constructs MST by sorting edges and using DSU to avoid cycles. 🌟
8. **🖍️ Flood Fill Algorithm**:
   * Fills a contiguous region in a grid or image with a specific color/value from a starting point. 🎨

Feeling accomplished and ready to tackle more challenges! 💪✨

#GraphTheory #Algorithms #Coding #StriverSheetChallenge #ProgrammingJourney #DSA

**Graph Revision Notes**

**Overview**

* **Graph**: A non-linear data structure consisting of nodes (vertices) connected by edges.
* **Applications**:
  + Modeling **electrical circuits**.
  + Finding the **shortest routes** between two locations (e.g., navigation systems like Google Maps).
  + **Social networks**, web page ranking (e.g., Google Search), and more.

**Key Concepts**

* **Degree of a Node**:
  + **Degree**: The number of edges connected to a node.
  + **In-Degree**: The number of edges coming into a node.
  + **Out-Degree**: The number of edges going out from a node.
* **Connected Components**:
  + **Connected Graph**: A graph in which there's a path between every pair of nodes.
  + **Disconnected Graph**: A graph that has more than one connected component (a set of nodes where each node is connected, but there's no path between nodes of different components).

**Graph Traversals**

* **Breadth-First Search (BFS)**:
  + Explores the graph layer by layer, starting from the root node (or any starting node), visiting all nodes at the present depth level before moving on to nodes at the next depth level.
  + Implemented using a **queue**.
* **Depth-First Search (DFS)**:
  + Explores as far as possible along each branch before backtracking.
  + Implemented using a **stack** (or recursion).

These traversal methods are foundational for exploring and analyzing graph structures.