1-Week Graph Mastery Plan

This document provides a structured 1-week plan to become proficient in graph theory and algorithms. It includes daily study goals, patterns, resources, real-case scenarios, and YouTube links for practical examples.

Graph Patterns & Concepts

- Graph representations: adjacency list, matrix, edge list
- Traversals: BFS, DFS
- Connectivity & components (connected, strongly connected)
- Trees and special graphs: spanning trees, minimum spanning tree (MST)
- Shortest paths: Dijkstra's, Bellman-Ford, Floyd-Warshall
- Topological sort / DAGs
- Network flows / max-flow min-cut
- Graph properties: cycles, bipartite, planarity
- Advanced topics: graph isomorphism, coloring, dynamic graphs

1-Week Schedule

Day 1: Basics: BFS, DFS, connectivity; Implement BFS/DFS; Example: Social network reachability

Day 2: Trees, DAGs, cycle detection, topological sort; Example: Task scheduling

Day 3: Shortest paths: Dijkstra, Bellman-Ford, Floyd-Warshall; Example: GPS routing

Day 4: Spanning Trees, MST (Prim's, Kruskal's); Example: Network design

Day 5: Network flows, matching, bipartite graphs; Example: Job assignment problems

Day 6: Advanced topics: dynamic graphs, graph in ML; Example: Knowledge graphs, GNNs

Day 7: Review, solve challenge problems, apply to real-world project

Resources

- Graph Algorithms for Technical Interviews – FreeCodeCamp (YouTube): https://www.youtube.com/watch?v=tWVWeAqZ0WU
- Graph Theory Tutorial from a Google Engineer (YouTube): https://www.youtube.com/watch?v=09_LlHjoEiY
- Graph Algorithms: Predict Real-World Behavior (YouTube): https://www.youtube.com/watch?v=dlL7WRFskTg
- Applications of Graph Algorithms – Memgraph: https://memgraph.com/blog/graph-algorithms-applications

- Introduction to Graph Theory – DataCamp: https://www.datacamp.com/tutorial/introduction-to-graph-theory

Real-World Applications

- Social networks: friend suggestions, influence spread (BFS/DFS, centrality)
- Transportation: GPS path-finding, logistics (Dijkstra, A*)
- Supply chains: max flow, min cut (Ford-Fulkerson, Edmonds-Karp)
- Task scheduling: dependency resolution (Topological sort)
- Recommender systems: knowledge graphs, embeddings
- Search engines: web crawling, PageRank