

DSA Project

Network Routing Optimization Using
Dijkstra's Algorithm

Meet the group

- 01 Soumya Sekhar Parida (B24CM1072)
- 02 Jiya Meena(B24EE1028)
- 03 Jahnavi Ballikuru (B24EE1010)

Contribution

- **01** Soumya Sekhar Parida (B24CM1072) -Dijkstra's Algorithm using priority queues and python visualisation of the graphs , GitHub repo structure, handling Overload
- **02** Jiya Meena(B24EE1028) - DFS Algorithm, Real world routing data into the graphs and in presentation slides, Handling Busy Nodes
- **03** Jahnvi Ballikuru (B24EE1010)- BFS Algorithm, Header file and declaring functions and implementing adjacency list , Handling Busy Nodes

Introduction

Why Routing Optimization Matters:

- Internet -> huge network of routers
- Every router connected by links
- Data must travel through the fastest path
- High delays means poor performance

Problem Statement

The Problem:

- Network links have different latencies
- Need the minimum-latency path
- Dynamic and complex network
- Manual selection impossible

Proposed Solution

Our Solution:

- Model routers as graph nodes
- Model connections as weighted edges
- Use Dijkstra's Algorithm
- Visualize path on a topology graph

HOW - WHAT - WHY

01

C++ Backend:

- Reads network file
- Builds graph
- Runs Dijkstra
- Saves shortest path + topology

02

Python Visualization:

- Loads CSV + path
- Draws full topology
- Highlights optimal path

03

Graph Structure:

- `adjacency[u]` = list of `(v, weight)`
- Undirected network
- Weight = latency (ms)

How Dijkstra Works:

- Uses min-heap priority queue
- Always picks router with minimum cost
- Updates neighbors
- Builds `parent[]` to reconstruct route

04

SPEICIAL FEATURE

1. We can make a node busy so that no data passes through it and the algorithm will take different path other than the shortest path.

2. We also see if a Wi-Fi router is having limits in sending data then also we can change path to send the information.

CODE WALKTHROUGH



DEMO

Real-World Applications

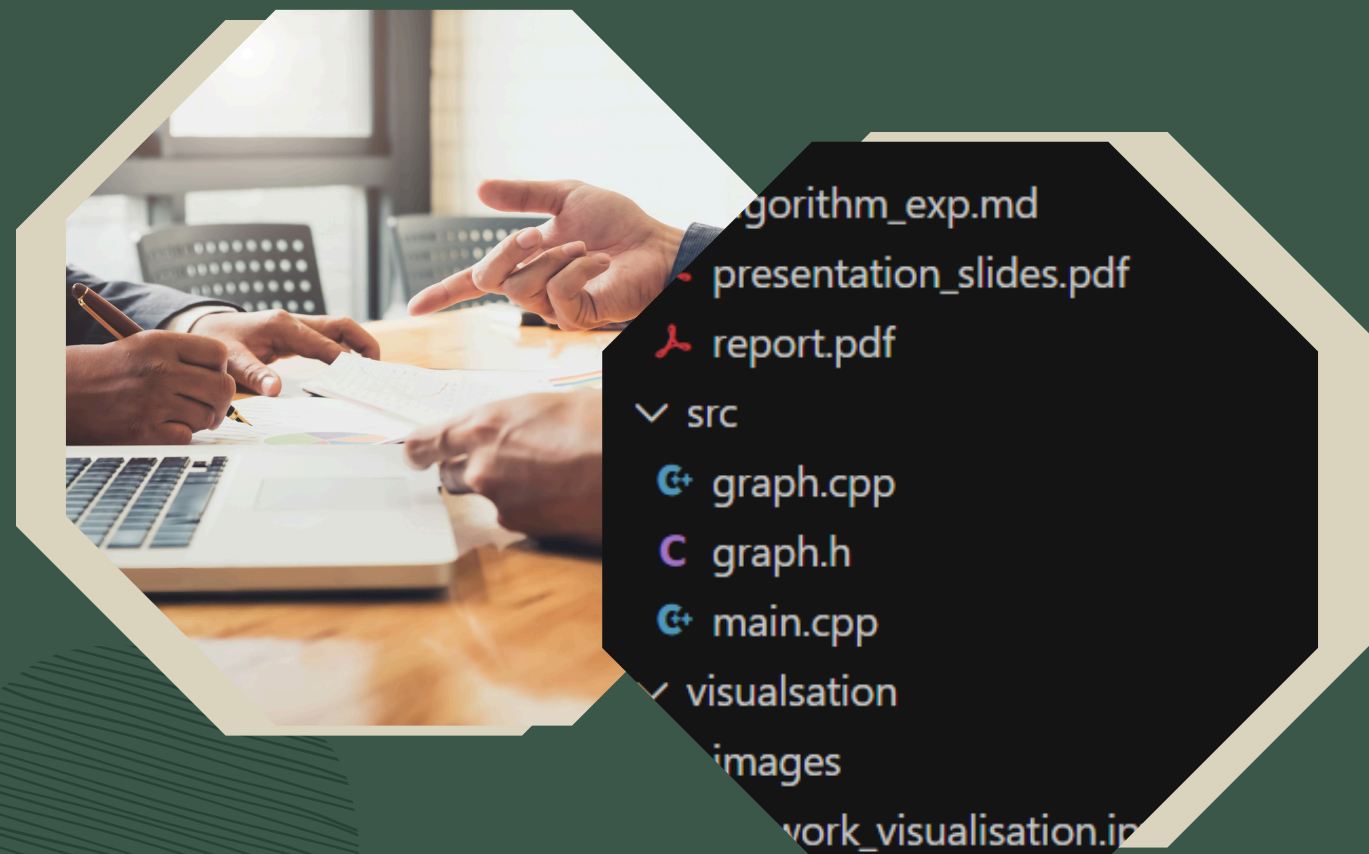
Used In:

- OSPF (Internet routing)
- Telecom backbone networks
- Data centers
- Cloud routing algorithms

Conclusion

Summary:

- Efficient routing simulation
- Models real Internet behavior
- Combines C++ & Python
- Demonstrates real-world DSA application





**Thank
You**