

Dijkstra? — [Dijkstra? Problem from Codeforces](#)

(Dijkstra): একটি weighted shortest path সমাধান

সমস্যা বিবরণ (Problem Description)

এই সমস্যা হলো weighted shortest path

তুমি একটি গ্রাফের একটি source node থেকে অন্য destination node-এ যেতে চাও।

গ্রাফের প্রতিটি edge-এর একটি ওজন (weight) আছে। লক্ষ্য:

সর্বনিম্ন মোট ওজন (minimum total weight) দিয়ে গন্তব্য node-এ পৌঁছানো।

এই ধরনের সমস্যা unweighted graph নয়, তাই BFS ব্যবহার করা যাবে না।

এখানে প্রয়োজন Dijkstra's Algorithm।

গ্রাফের বিবরণ (Graph Details)

- গ্রাফটি হতে পারে directed বা undirected
- N node, M edge
- $edge[i] = (u, v, w)$ যেখানে $w =$ ওজন
- edge-এর weight ধনাত্মক (positive)
- লক্ষ্য হলো source \rightarrow destination সর্বনিম্ন ওজনের পথ খুঁজে বের করা

কেন Dijkstra? (Why Dijkstra?)

Dijkstra অ্যালগরিদম একটি greedy shortest path algorithm।

এটি প্রতিটি node-এর current known shortest distance ট্র্যাক করে,

priority queue ব্যবহার করে সবচেয়ে কম distance node প্রথমে প্রসেস করে।

- Layer বা BFS-এর মতো sequential নয়
- বরং সর্বদা current shortest path অনুযায়ী প্রসেস করা হয়
- সঠিকভাবে ওজনের সঙ্গে shortest path নিশ্চিত করা হয়

সমাধানের বৈশিষ্ট্য (Features)

- Weighted graph-এর জন্য উপযুক্ত
- Priority Queue ব্যবহার করে efficient solution
- Directed এবং undirected উভয় গ্রাফেই কাজ করে
- Edge weight ধনাত্মক হলে সর্বদা সঠিক ফলাফল

Pseudocode

```
FUNCTION dijkstra(graph, start, n):  
    DEFINE dist[1..n] = INFINITY  
    DEFINE visited[1..n] = False  
    DEFINE priority_queue pq  
  
    dist[start] = 0  
    PUSH (0, start) INTO pq  
  
    WHILE pq IS NOT EMPTY:  
        (d, u) = POP pq  
        IF visited[u]:  
            CONTINUE  
        visited[u] = True  
  
        FOR each (v, w) in graph[u]:  
            IF dist[u] + w < dist[v]:  
                dist[v] = dist[u] + w  
                PUSH (dist[v], v) INTO pq  
  
    RETURN dist
```