PCTE-IET

B.Tech CSE 4th SEM

DESIGN AND ANALYSIS OF ALGORITHMS

(BTCS -403-18)

PRESENTATION SYNOPSIS

TOPIC: All Source Shortest Path

NAME: Harshit Bansal , Harsimranjit Kaur , Harleen Kaur , Himanshu

UNIV.ROLL.NO: 2230016 , 2230018 , 2230015 , 2230020

OFFICIAL E-MAIL ADDRESS: harshitbansal184507@gmail.com , harsimran12003@gmail.com , Harleen242004@gmail.com , himanshudeolia0825@gmail.com

TABLE OF CONTENTS :

BRIEF INTRODUCTION OF TOPIC:-

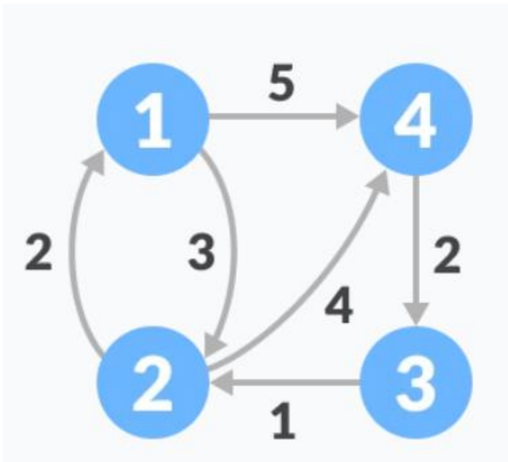
Floyd-Warshall Algorithm is an algorithm for finding the shortest path between all the pairs of vertices in a weighted graph. This algorithm works for both the directed and undirected weighted graphs. But, it does not work for the graphs with negative cycles (where the sum of the edges in a cycle is negative). This algorithm follows the dynamic programming approach to find the shortest paths.

KEY APPLICATION AREAS OF TOPIC IN REAL LIFE:

- To find the shortest path in a directed graph
- To find the transitive closure of directed graphs
- To find the Inversion of real matrices
- For testing whether an undirected graph is bipartite

LOGICAL EXPLANATION OF TOPIC / BLOCK DIAGRAM :

GIVEN GRAPH (Reference for synopsis only)



$$A^0 = \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & 0 & 3 & \infty & 5 \\ 2 & 2 & 0 & \infty & 4 \\ 3 & \infty & 1 & 0 & \infty \\ 4 & \infty & \infty & 2 & 0 \end{array}$$

$$A^1 = \begin{array}{cc} & \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \end{array} & \begin{bmatrix} 0 & 3 & \infty & 5 \\ 2 & 0 & & \\ \infty & & 0 & \\ \infty & & & 0 \end{bmatrix} \end{array} \rightarrow \begin{array}{cc} & \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \end{array} & \begin{bmatrix} 0 & 3 & \infty & 5 \\ 2 & 0 & 9 & 4 \\ \infty & 1 & 0 & 8 \\ \infty & \infty & 2 & 0 \end{bmatrix} \end{array}$$

$$A^2 = \begin{array}{cc} & \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \end{array} & \begin{bmatrix} 0 & 3 & & \\ 2 & 0 & 9 & 4 \\ & 1 & 0 & \\ & \infty & & 0 \end{bmatrix} \end{array} \rightarrow \begin{array}{cc} & \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \end{array} & \begin{bmatrix} 0 & 3 & 9 & 5 \\ 2 & 0 & 9 & 4 \\ 3 & 1 & 0 & 5 \\ \infty & \infty & 2 & 0 \end{bmatrix} \end{array}$$

$$A^3 = \begin{array}{cc} & \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \end{array} & \begin{bmatrix} 0 & & \infty & \\ & 0 & 9 & \\ \infty & 1 & 0 & 8 \\ & & 2 & 0 \end{bmatrix} \end{array} \rightarrow \begin{array}{cc} & \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \end{array} & \begin{bmatrix} 0 & 3 & 9 & 5 \\ 2 & 0 & 9 & 4 \\ 3 & 1 & 0 & 5 \\ 5 & 3 & 2 & 0 \end{bmatrix} \end{array}$$

$$A^4 = \begin{array}{cc} & \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \end{array} & \begin{bmatrix} 0 & & & 5 \\ & 0 & & 4 \\ & & 0 & 5 \\ 5 & 3 & 2 & 0 \end{bmatrix} \end{array} \rightarrow \begin{array}{cc} & \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \end{array} & \begin{bmatrix} 0 & 3 & 7 & 5 \\ 2 & 0 & 6 & 4 \\ 3 & 1 & 0 & 5 \\ 5 & 3 & 2 & 0 \end{bmatrix} \end{array}$$

Iterations according to the algorithm and the result is stored in A4 .

- **Time Complexity**

There are three loops. Each loop has constant complexities. So, the time complexity of the Floyd-Warshall algorithm is $O(n^3)$.

- **Space Complexity**

The space complexity of the Floyd-Warshall algorithm is $O(n^2)$.

LIMITATIONS:

- The algorithm has a high time complexity of $O(V^3)$, where V is the number of vertices in the graph.
- The algorithm requires a large amount of memory to store the distance matrix.
- The algorithm does not work well for very large graphs.

CONCLUSION:

In summary, the Floyd-Warshall algorithm offers a straightforward yet powerful solution to finding the shortest paths between all pairs of vertices in a weighted graph. Its efficiency and simplicity make it a valuable tool in various applications such as network routing and traffic management. With a time complexity of O(V^3), it's well-suited for moderately sized graphs and is easily implementable. Overall, Floyd-Warshall stands as a fundamental concept in graph theory, demonstrating the elegance and practicality of algorithmic design.

FULL URL OF RESOURCES:

1) https://www.programiz.com/dsa/floyd-warshall-algorithm
2) https://takeuforward.org/data-structure/floyd-warshall-algorithm-g-42/