EXAM SHEDULING USING MAXIMUM FLOW PROBLEM

(DINIC'S ALGORITHM)

PREPARED BY

MUHAMMAD JAWAD AMIN

0175-BSCS-2020

Section (A)

SUBMITTED TO

SIR SAEED-U-REHMAN TURK

BACKGROUND:

In optimization theory, **maximum flow problems** involve finding a feasible flow through a flow network that obtains the maximum possible flow rate. The maximum flow problem was first formulated in 1954 by T. E. Harris and F. S. Ross as a simplified model of Soviet railway traffic flow. In 1955, Lester R. Ford, Jr. and Delbert R. Fulkerson created the first known algorithm, the Ford–Fulkerson algorithm. In their 1955 paper, Ford and Fulkerson wrote that the problem of Harris and Ross is formulated as follows:

"Consider a rail network connecting two cities by way of several intermediate cities, where each link of the network has a number assigned to it representing its capacity. Assuming a steady state condition, find a maximal flow from one given city to the other."

Yefim Dinitz invented this algorithm in response to a pre-class exercise in Adelson-Velsky's algorithms class. At the time he was not aware of the basic facts regarding the Ford–Fulkerson algorithm. Dinitz mentions inventing his algorithm in January 1969, which was published in 1970 in the journal Doklady Akademii Nauk SSSR. This algorithm is also known as Dinic's Algorithm for Maximum Flow.

APPLICATIONS:

Dinic's Maximum Flow algorithm is used to solve the different maximum flow problem in the real life which are as follows:

- Airline scheduling
- Exam scheduling
- Maximum water flow through pipes
- Maximum current Flow
- Maximum packets flow through a network

- Maximum traffic flow through a map.
- Bipartite Matching
- Tuple Selection
- And many more…….

PROBLEM STATEMENT:

Given a directed graph which represents a flow network where every edge has a capacity. Also given two vertices source 's' and sink 't' in the graph, find the maximum possible flow from s to t with following constraints:

- Flow on an edge doesn't exceed the given capacity of the edge.
- Incoming flow is equal to outgoing flow for every vertex except s and t.

The algorithm runs in time O (V^2E), as it uses shortest augmenting paths. It working at is also very efficient for the bipartite graphs with the complexity of O($\sqrt{V}E$). It is also more efficient than the Ford Fulkerson Algorithm.

OBJECTIVES:

By using this algorithm, we can solve the many maximum flow problems. I will solve the Exam Scheduling Problem which every Educational Institute has to face when they schedule the Exams. The problems which they faced are as follows:

- How many rooms are available?
- How many proctors are there to handle the students?
- What are the timings for the papers?
- How many classes are which are participating in the exams?

I am just mentioning the problem because we must be specific. I want to choose such a problem by which I could learn the Maximum Flow Problem in depth.

REFERENCES:

https://en.wikipedia.org/wiki/Dinic%27s_algorithm

https://en.wikipedia.org/wiki/Maximum_flow_problem

https://www.geeksforgeeks.org/dinics-algorithm-maximum-flow/