ALGORITHMS AND PROBLEM SOLVING LAB

PROJECT SYNOPSIS

Abhishek Verma	16103237
Akshanshu Kumar	16103112
Shiva Gupta	16103094
Tushar Jain	16103198

THE GRIM REAPER ADVANCED MILITARY ARSENAL (GRAMA)

THE FOLLOWING PROJECT HAS BEEN IMPLEMENTED IN C++;

OBJECTIVE:-

This project is aimed at computing minimum amounts of ammunition required to disrupt a country's networking of roads and cities. This military arsenal gives us the roads and cities which must be destroyed in order to cut-off transfer of goods/ oil/ ammunition/ soldiers etc. in a country, thus dividing it and making it easy to conquer. This software also gives us the maximum amount of goods/ oil/ ammunition /soldiers etc. that can be transferred from a source city to a destination city thus giving us an estimate of enemy nations power.

ALGORITHMS/ DATA STRUCTURES USED :-

- Graphs
- Articulation Points
- Bridges
- Maximum Flow
- Ford Fulkerson
- Standart Template Library of C++

WORKING:-

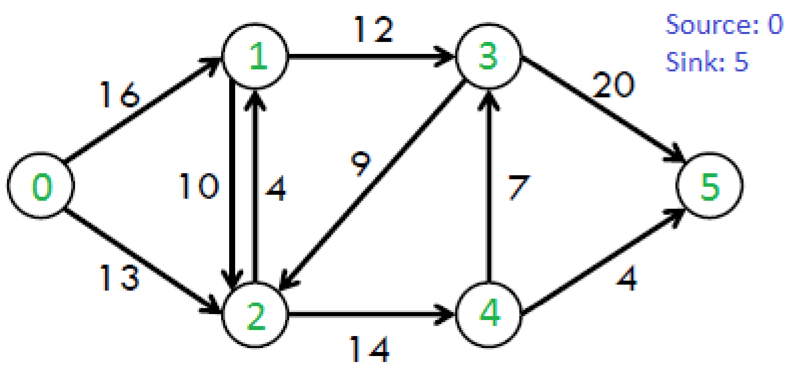
The country map is implemented as a graph and each city serves as a vertex and each road connecting the cities serves as an edge. The Ford Fulkerson Algorithm implemented to find the Maximum Flow via the graph will give us the amount of maximum goods/ oil/ ammunition/soldiers etc. which can be carried from a source city (source vertex) to destination city (destination vertex) which could help in estimating the enemy's power . The articulation points will give us the cities which must be destroyed thus resulting in disruption of goods/ oil/ ammunition/soldiers etc. and bridges will give us the roads which are connecting cities and must be destroyed thus only requiring the minimum battles to be fought to win the war. There is no point in attacking every city so this arsenal gives us an efficient and smart approach to give the enemies the taste of defeat!

MAXIMUM FLOW

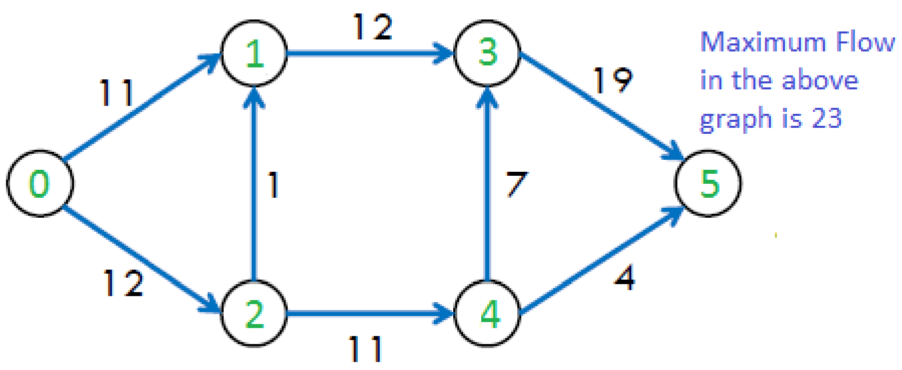
Given a graph which represents a flow network where every edge has a capacity. Also given two vertices *source* 's' and *sink* 't' in the graph, find the maximum possible flow from s to t with following constraints:

a) Flow on an edge doesn't exceed the given capacity of the edge.

b) Incoming flow is equal to outgoing flow for every vertex except s and t.



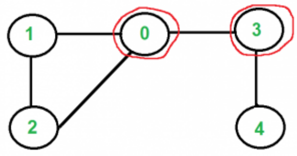
The maximum possible flow in the above graph is 23.



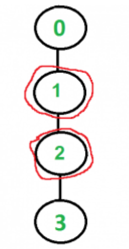
ARTICULATION POINT

A vertex in an undirected connected graph is an articulation point (or cut vertex) iff removing it (and edges through it) disconnects the graph. Articulation points represent vulnerabilities in a connected network – single points whose failure would split the network into 2 or more disconnected components. They are useful for designing reliable networks.
For a disconnected undirected graph, an articulation point is a vertex removing which increases number of

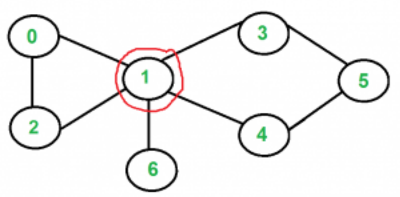
connected components.
Following are some example graphs with articulation points encircled with red color.



Articulation points are 0 and 3



Artiulation
Points are 1 & 2



Articulation Point is 1

BRIDGES

An edge in an undirected connected graph is a bridge iff removing it disconnects the graph. For a disconnected undirected graph, definition is similar, a bridge is an edge removing which increases number of connected components.
Like Articulation Points, bridges represent vulnerabilities in a connected network and are useful for designing reliable networks. For example, in a wired computer

network, an articulation point indicates the critical computers and a bridge indicates the critical wires or connections.

Following are some example graphs with bridges highlighted with red color.