

**Project REPORT**

**SPRING 2021**

**Abeeha Fatima 19i-0742 (CS-E)**

**Amna Shafiq 19i-1978 (CS-E)**

**Faizan Shabbir 19i-2027 (CS-C)**

**CS\_ 218 DATA STRUCTURES**

**Submission date: 20th June 2021**

**Instructor: Sir Hassan Raza**

Dataset that we have chosen is “ca-QrGC.txt.gz”. Since, it is the smallest one amongst all of the sets and was taking less time to compile on my laptop because it was taking less memory.

Apart from this the other datasets were way too large that is why the complexity of the data was increased so in order to avoid that I chose dataset which was comparatively smaller than the others.

The basic data structures that we used in our project is of Adjacency Link List. The problem we faced during this project was the implementation of BFS. Since, one of our group members was working with linked list and the other one was working on BFS. And while working on BFS we used Arrays. So, in order to solve this issue, we used OPERATOR OVERLOADING in our link list and overloaded the Index. And through this we were able to solve this issue and used our Arrays simultaneously with linked list.

In first set, we used basic simple counter to count the nodes being created. Similarly, when the edges were created, we placed another counter which recorded the number of edges being created in the graph. While, finding the articulation nodes of the graph we used BFS.

To find sink, source node and isolated nodes in the graph we created two adjacency lists for outward connections and inward connections respectively. To find the number of source nodes, in the inward connection list of each vertex, we check if there is a zero. If zero is present, we increment the counter. Similarly, to find the number of sink nodes, we check if there is a zero in outward connection list. If yes, we increment the counter set for sink nodes. For isolated nodes, if both in source and out source of a vertex is empty, we count it as isolated node. In this way, when we checked in the given data   
set, number of all three (sink nodes, source nodes and isolated nodes) is 0. Also, for remaining points of SCC we used its algorithm and for WCC we used its algorithm.

Sheet: <https://docs.google.com/spreadsheets/d/1-QBIxcZlCXfn08wsxpEp-H3pf5DUwpNxl-mT9NTaBc8/edit?usp=sharing>