



VIETNAM NATIONAL UNIVERSITY, HCMC
UNIVERSITY OF SCIENCE
FACULTY OF INFORMATION TECHNOLOGY

DATA STRUCTURES AND ALGORITHMS

PROJECT REPORT – 19CLC3

CHALLENGE 3:

COUNT STARS

[LECTURERS]

- PhD. Nguyễn Hải Minh
- M.Sc. Bùi Huy Thông
- M.Sc. Trần Thị Thảo Nhi

[GROUP 41469598]

- | | |
|----------------------|----------|
| - Lê Thị Phương Linh | 19127041 |
| - Trần Văn Quý Phước | 19127246 |
| - Nguyễn Minh Trí | 19127595 |
| - Lê Yến Nhi | 19127498 |

1

WORKING PROGRESS

Group Information

Student ID	Full Name	Email	Roles
19127041	Lê Thị Phương Linh	19127041@student.hcmus.edu.vn	Leader
19127246	Trần Văn Quý Phước	19127246@student.hcmus.edu.vn	Developer
19127595	Nguyễn Minh Trí	19127595@student.hcmus.edu.vn	Tester
19127498	Lê Yến Nhi	19127498@student.hcmus.edu.vn	Secretary

Project Information

Project name	COUNT STARS	
Tools and functions	Messenger	Communicate each other
	CodeBlocks	Coding
	Google Documents, Office 365	Write report and functions
	ZOOM Cloud Meetings	Meeting

Meeting

1. Overview

Group name: Group 41469598

Members: Lê Thị Phương Linh, Trần Văn Quý Phước, Nguyễn Minh Trí, Lê Yến Nhi.

Purpose of the meeting:

1. Discuss about the project.
2. Get the division of works.

Place: ZOOM Cloud Meetings, Messenger.

Date: Tuesday 15/12/2020

Time: 9:30am – 11am

Status: 100% DONE

2. Table of works

ID	NAME	DESCRIPTION	STATUS
19127041	Lê Thị Phương Linh	- Project Manager - Coder	DONE
19127246	Trần Văn Quý Phước	- Developer	DONE
19127595	Nguyễn Minh Trí	- Coder - Tester	DONE
19127498	Lê Yến Nhi	- Write report - Summarize the project	DONE

2

INFORMATION OF CODE FRAGMENTS

2.1 Data structures

Class template

```
std::queue
template <class T, class Container = deque<T> > class queue;

std::vector
template < class T, class Alloc = allocator<T> > class vector;
// generic template
```

2.2 Algorithms

```
int countStars(vector <vector<bool>> adj)
```

Divide the original graph into connected components. In turn, check the subgraphs generated by those connected components is a star or not.

After each check, delete that connected component.

```
bool checkStarGraph(vector <vector<bool>> adj)
```

If number of vertices having degree 1 is $n-1$ and number of vertex having degree $n-1$ is 1 then our graph should be a star graph otherwise not a star graph.

```
void removeVertex(int pos, vector <vector<bool>> adj)
```

Firstly, we check if that vertex exists in the graph or not. If that vertex exists then we need to shift the rows to the left and the columns upwards of the adjacency matrix so that the row and column values of the given vertex gets replaced by the values of the next vertex and then decrease the number of vertices by 1. In this way that particular vertex will be removed from the adjacency matrix.

```
vector <vector<short>> generateAdjList(vector <vector<bool>> adj)
```

Create an array of lists and traverse the adjacency matrix. If any adj[i][j] in the matrix = 1. It means there is an edge from i to j, so insert j in the list at i-th position in the array of lists.

```
vector <vector<bool>> buildAdjFromConnectedComponent
    (vector <short> vertex, vector <vector<bool>> adj)
```

Initialize a tempAdj to store result, temp to store a row of matrix. If for any cell vertex[i] and vertex[j] at the adj matrix is 1 then push 1 into temp else push 0 into temp. After a loop, push temp in to tempAdj.

2.3 Functions

- Enter Adjacency matrix.

```
vector <vector<bool>> InputAdjMatrix(int size)
```

- Print Adjacency matrix.

```
void OutputMatrix (vector <vector<bool>> adj)
```

- Check input matrix is a star graph .

```
bool checkStarGraph(vector<vector<bool>> adj)
```

- Remove vertex "pos" from graph.

```
void removeVertex(int pos, vector<vector<bool>> &adj)
```

- Check for adjacency of 2 vertices of the graph.

```
bool isAdjacency(vector<vector<bool>>adj, int v1, int v2)
```

- Generate an adjacency list.

```
vector<vector<short>> generateAdjList(vector<vector<bool>> adj)
```

- Build an adjacency matrix from connected component.

```
vector <vector<bool>> buildAdjFromConnectedComponent  
    (vector <short> vertex, vector<vector<bool>> adj)
```

- Count stars.

```
int countStars(vector <vector<bool>> adj)
```

3

REFERENCES

<http://www.cplusplus.com/reference/>

<https://www.geeksforgeeks.org/check-star-graph/?fbclid=IwAR0Tzn1aH0G43Ues-t32YG247U3e1SU2Uhx11Z-K8O4uztLleXN5o8LjM8>

<https://www.geeksforgeeks.org/add-and-remove-vertex-in-adjacency-matrix-representation-of-graph/>