# CSCE 2110.001
# Project 3:

***Contributors:***

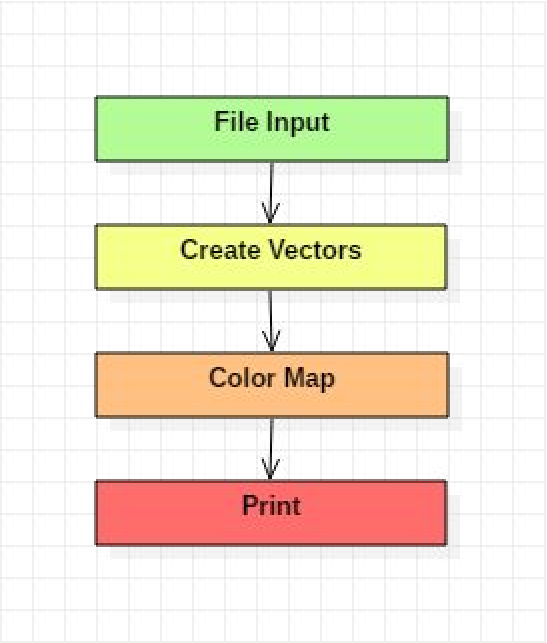
***Read Ballew***

***Paris Estes***

***Thien-An Vu***

***Due December 5th, 2018***

# Design Process



## 1 ) Design Process

The first step of our design process was writing down what steps need to be executed in order to complete the project. Then we proceeded to refine the design to only include what is necessary, and completed a StarUML chart detailing the final choices before we began coding. We did not create any small prototype for this project, it was not needed. We decided on the coding style ahead of time, to simplify naming variables and creating functions. The data structure types we most commonly used were: vectors, strings, and integers. We made the correct choice in our opinion, because the project was simple to follow logically thanks to the sound data structures we created.

## 2 ) Data Structures

Most data overall is stored in vectors of strings, vector of vectors of integers, vector of vectors of strings (the adjacency list) and strings/ints/etc. Once the data is read in, we used the getline function, a string clean function (self-created), and a number clean function(self-created) to parse it and further check each component of the input file. Once it was parsed/checked, we directed the data through the appropriate channels (functions). So that way, the files did not need to remain open while the program was running. We used two main vectors to control the data from the input, and parse that to further call each function. A vector of strings for the list of country names, vector of integers for the adjacency matrix, and a vector of vector of strings for the adjacency list.

We created a vector that contained a vector of strings. Each element of the outside vector refers to a country's adjacent countries. This contained the parsed data from each line in the adjacency matrix. Inside the main.cpp we had several vectors of strings and integers to maintain and organize all streams of information that we gleaned from the input files. We created the adjacency list by making a vector of vector of integers for the matrix and then turned that into the adjacency list using the vector of strings of country names.

## 3 ) System Functionality

The program pulled in all the data from the text file and filtered it into two different vectors, one for the continents and one for the connections themselves.

After that the program creates the adjacency list from the matrix. Then we call the different functions to calculate the degree and color the map

The color map function takes in all the data we have collected and scrolls through the adj list to check if a connection has been colored, if no connections have been colored the current country gets the first color in our color vector (which stores all the colors we can use). However if it detects a color it goes and checks if the next color is in the list, if it is not in the list then it assigns that county that color. This function returns the degree so that when we call the degree function to output the degree data it has the correct data.

## 4 ) Group Member Work Breakdown

The work was divided evenly between all the group members. Thien-An worked on reading in and storing the information, more specifically store them in their appropriate vectors in order to work with them later. He also worked on getting the degree of each countries utilize the stored input. There are a few but mostly just to store the country list and the countries each country comes in contact. Read worked on refine the strings that were pulled in and get rid of the commas and the bonus (color the text). He worked on the countries list string to delimit commas to make the data easier to work with. It's the same thing for the bit vector for the countries. For the bonus, he used the find function to find the string that matches the color and made them have colors on the terminal. Paris worked on the map coloring, which is the main part of the project. He initially set the ocean to be blue and anything that touches the ocean has to be to different color. After that, he checked the rest of the country in the countries, color and made sure no 2 countries next to each other have the same color.