



CSCI 2270 – Data Structures

Instructors: Christopher Godley, Maciej Zagrodzki

Assignment 8 - Graph

OBJECTIVES

1. Applications of BFS

Overview

In this assignment, you will apply BFS for finding connected cities in a graph and check whether a graph is Bipartite graph or not.

Graph Class

Your code should implement graph traversal for cities. A header file that lays out this graph can be found in [Graph.hpp](#) on Moodle. *As usual, do not modify the header file. You may implement helper functions in your .cpp file if you want as long as you don't add those functions to the Graph class.*

Your graph will utilize the following struct:

```
struct vertex;

struct adjVertex{
    vertex *v;
};

struct vertex{
    vertex() {
        this->visited = false;
        this->color = "";
        this->distance = 0;
    }
    string name;
    bool visited;
    string color;
    int distance;
    vector<adjVertex> adj;
};
```

void addVertex(string name);

→ Add new vertex 'name' to the graph.

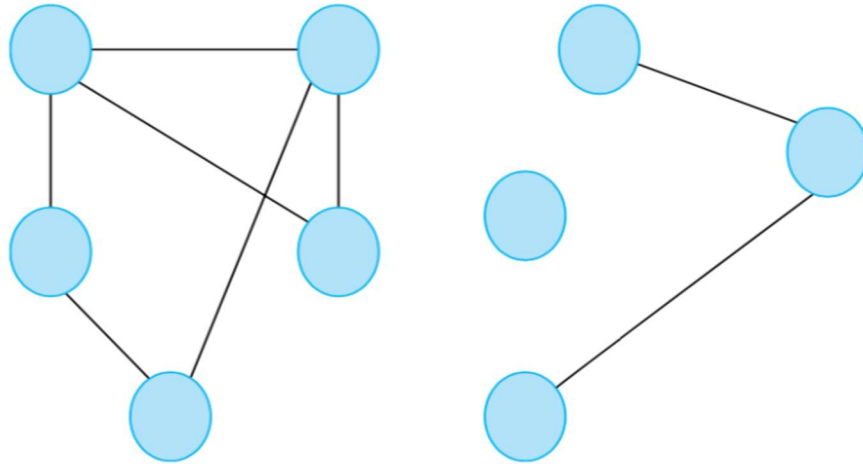
int getConnectedComponents();



CSCI 2270 – Data Structures

Instructors: Christopher Godley, Maciej Zagrodzki

- This method will provide the number of connected components in the graph i.e. the number of distinct subgraphs where no edges exist which connect vertices in two distinct subgraphs. In the following graph, the number of connected components is 3.



bool checkBipartite();

- This method will check whether a graph is bipartite or not. Will return true if the graph is Bipartite, otherwise false.

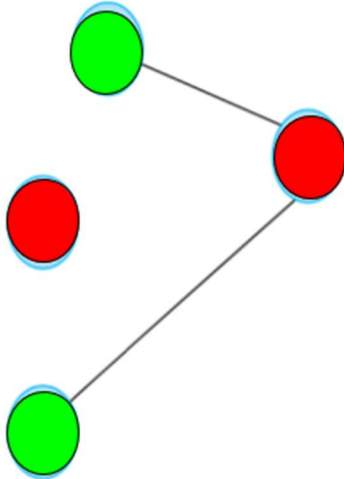
A [Bipartite graph](#) is a graph whose vertices can be divided into two sets and there will not be any edge among the vertices in the same set. We can say a graph is bipartite, if we can color the graph with two colors where vertices in the same set will be same colored.

In other words, in a Bipartite graph, no two adjacent vertices will have the same color.



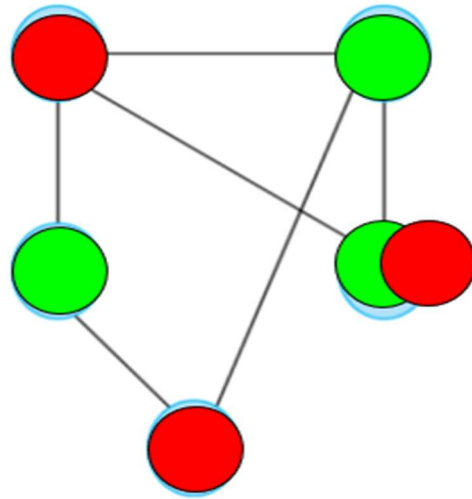
CSCI 2270 – Data Structures

Instructors: Christopher Godley, Maciej Zagrodzki



this graph, we color the vertices with two colors, so, it's a bipartite graph.

In



In this graph, if we try to color the vertices with two colors, we can't do that. As you can see, one vertex has one neighbor with the color red and the other with green. Therefore, for the graph to be bipartite, that node would have to be two colors at the same time. So, it's not a bipartite graph.

Please note that once you are done with your assignment on code runner you need to click on **'finish attempt'** and the **'submit all and finish'**. If you don't do this, you will not get graded."