Algorithm Archives

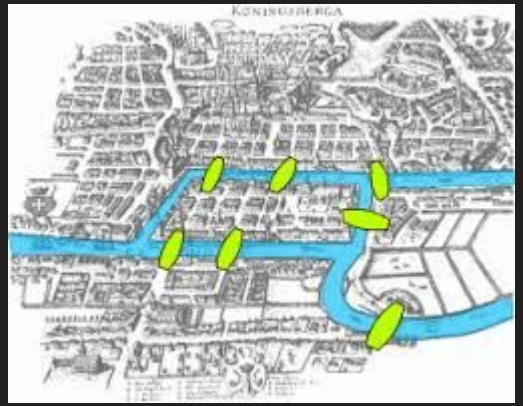
Gilbert Neuner, Daryan Sugandhi, Esam Izzat, Andrew Lybianto

Content

1. Graph Theory Overview
2. Hierarchy and Implementation
3. Let’s Play!

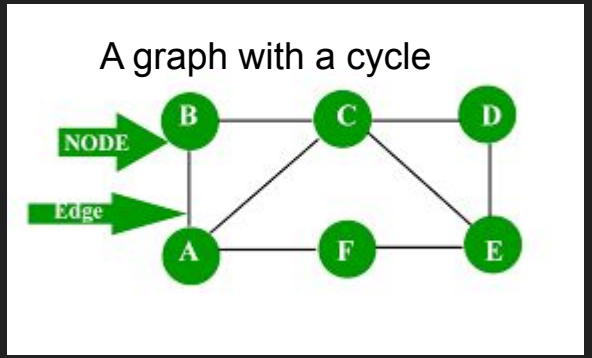
Project Overview

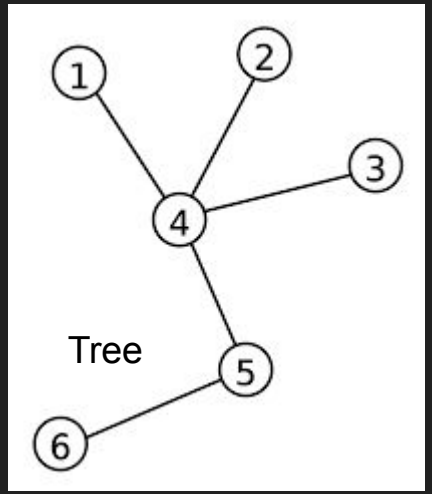
This project executes Depth First Search and Breadth First Search on a graph inputted by the user



Definitions

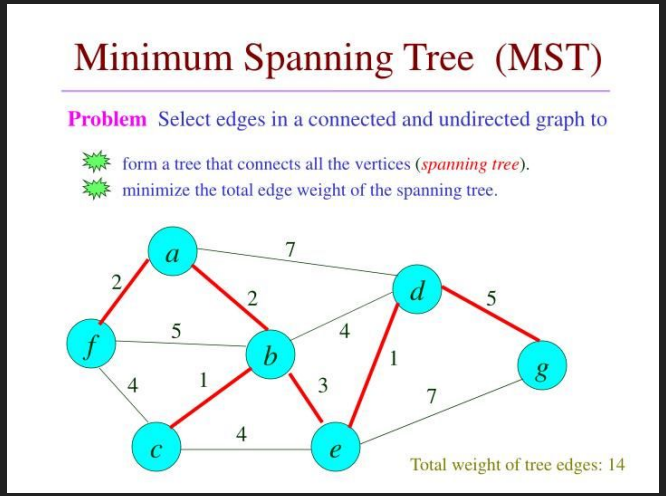
- A graph is a collection of nodes connected by edges
- A cycle is a set of vertices connected in a closed chain
- A tree is a connected graph with no cycles





Minimum Spanning Tree

- A minimum spanning tree is a subset of the edges of a graph which connects all the nodes together with no cycles
- MSTs are not always unique and different algorithms will yield different MSTs given the same inputs

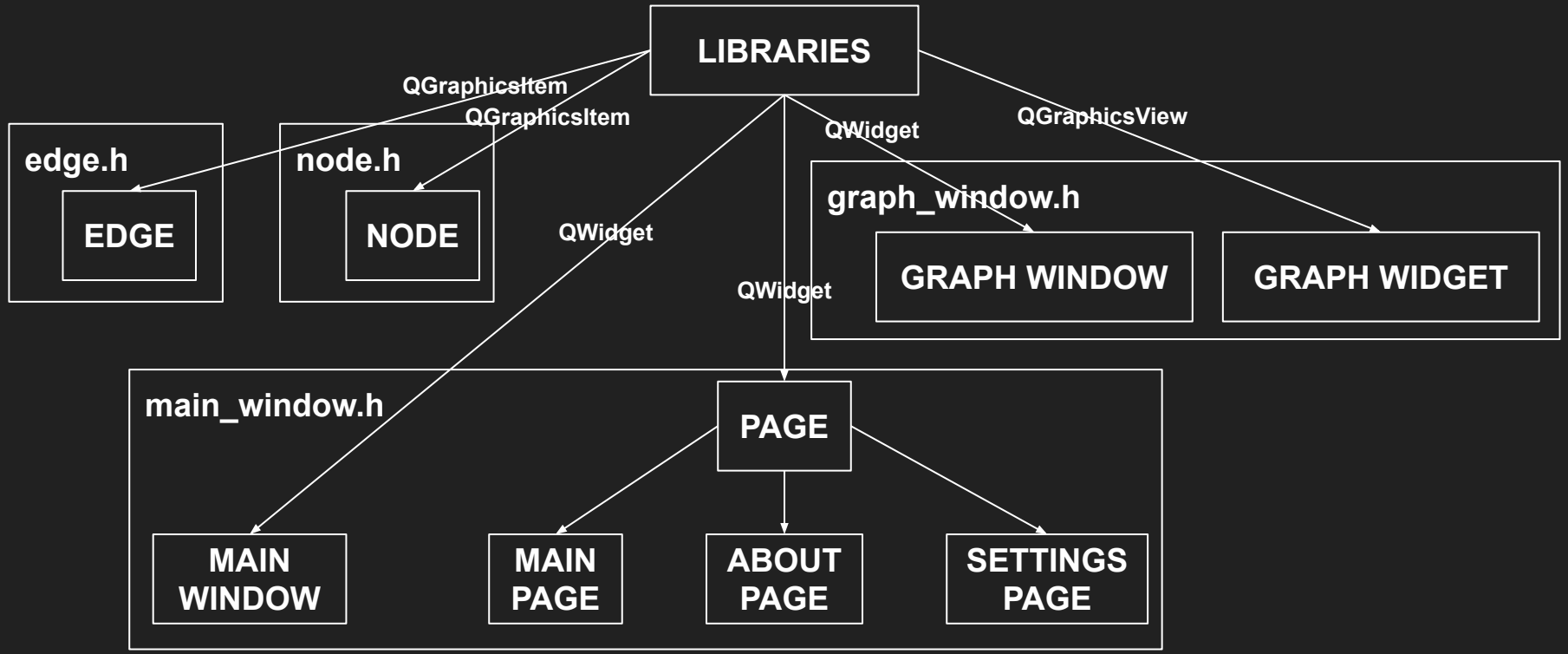


DFS vs BFS

- Depth First Search
- Finds MSTs by starting at root node, and exploring as far as possible before backtracking

- Breadth First Search
- Starts at root node, and explores all neighbors at current depth before moving onto nodes at the next depth

Class Hierarchies



Edge, Node, and Graph

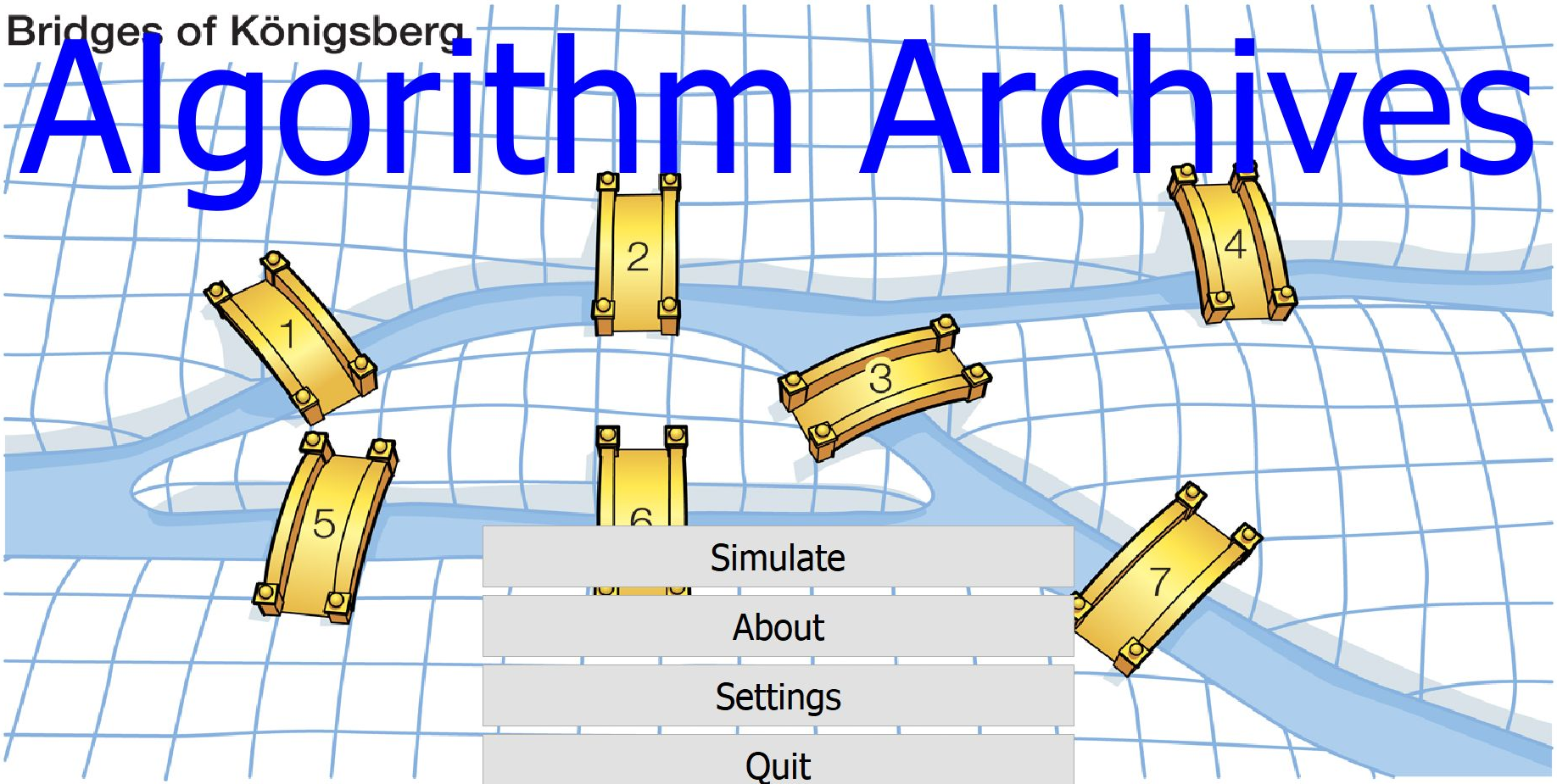
- Edge
  - Stores pointers to graph, and two nodes
  - Mostly cosmetic - nodes do the heavy lifting
- Node
  - Stores a pointer to graph, a vector of neighbors, a bool indicating whether it has been explored, and an identification number
  - Use this to look at other nodes
  - Use this to add and delete edges
- Graph
  - Stores vector of nodes, and a vector of edges
  - Also knows which identification numbers are available
  - Use this to add and delete nodes

Main Window/Misc. Class Overview

- MainWindow: window that links the simulation, about and settings page
- Page: parent class of the pages; return the contents and buttons of the page
- Main Page: title page
- About Page: descriptions of the algorithm and what they do
- Settings Page: allows the music to be muted/unmuted
- StackedWidget

Bridges of Königsberg

Algorithm Archives



Simulate

About

Settings

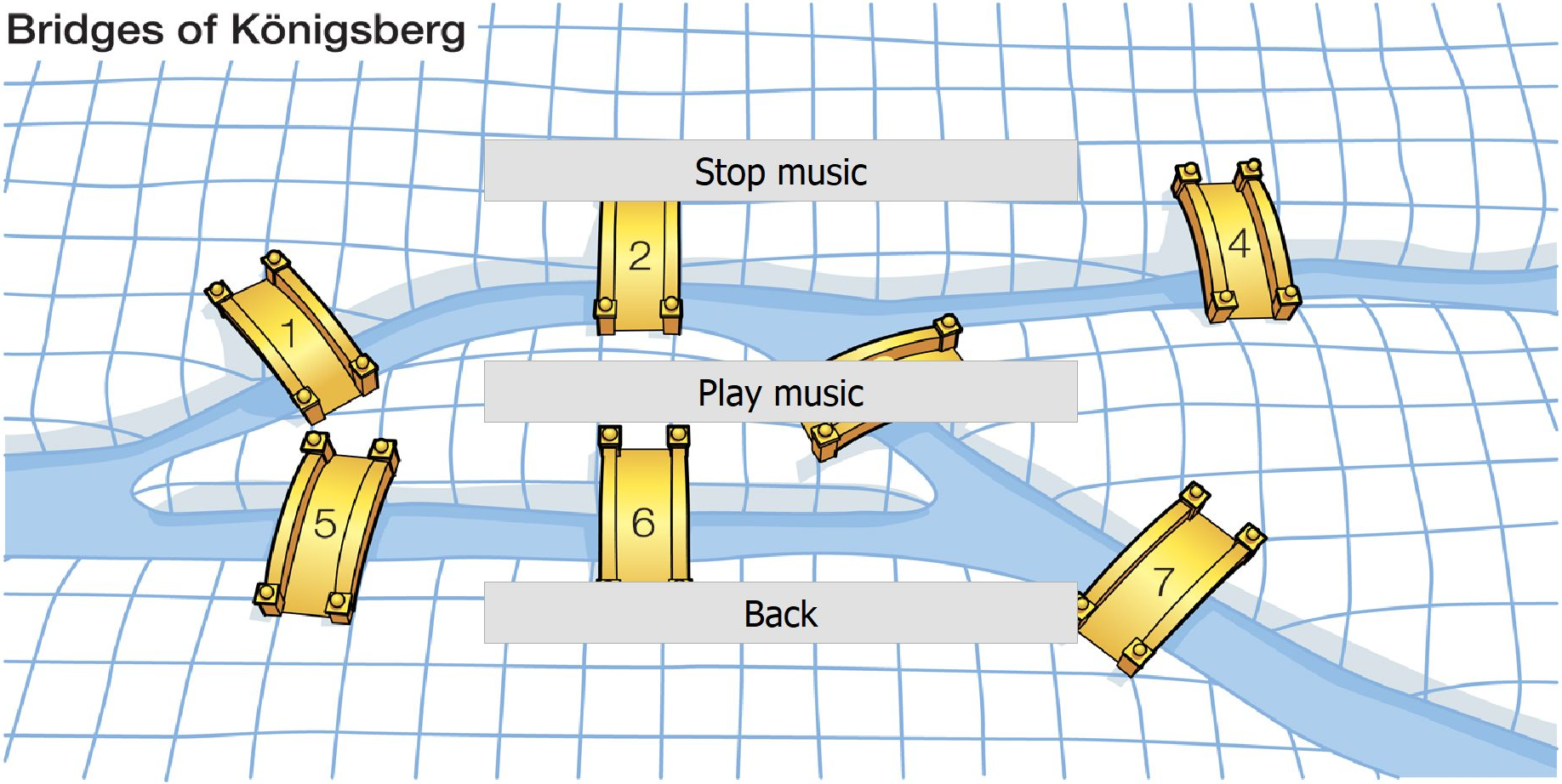
Quit

Programmers: Gilbert Neuner, Daryan Sugandhi, Esam Izzat, Andrew Lybianto

Description: This project executes Depth First Search and Breadth First Search on a graph inputted by the user.

Back

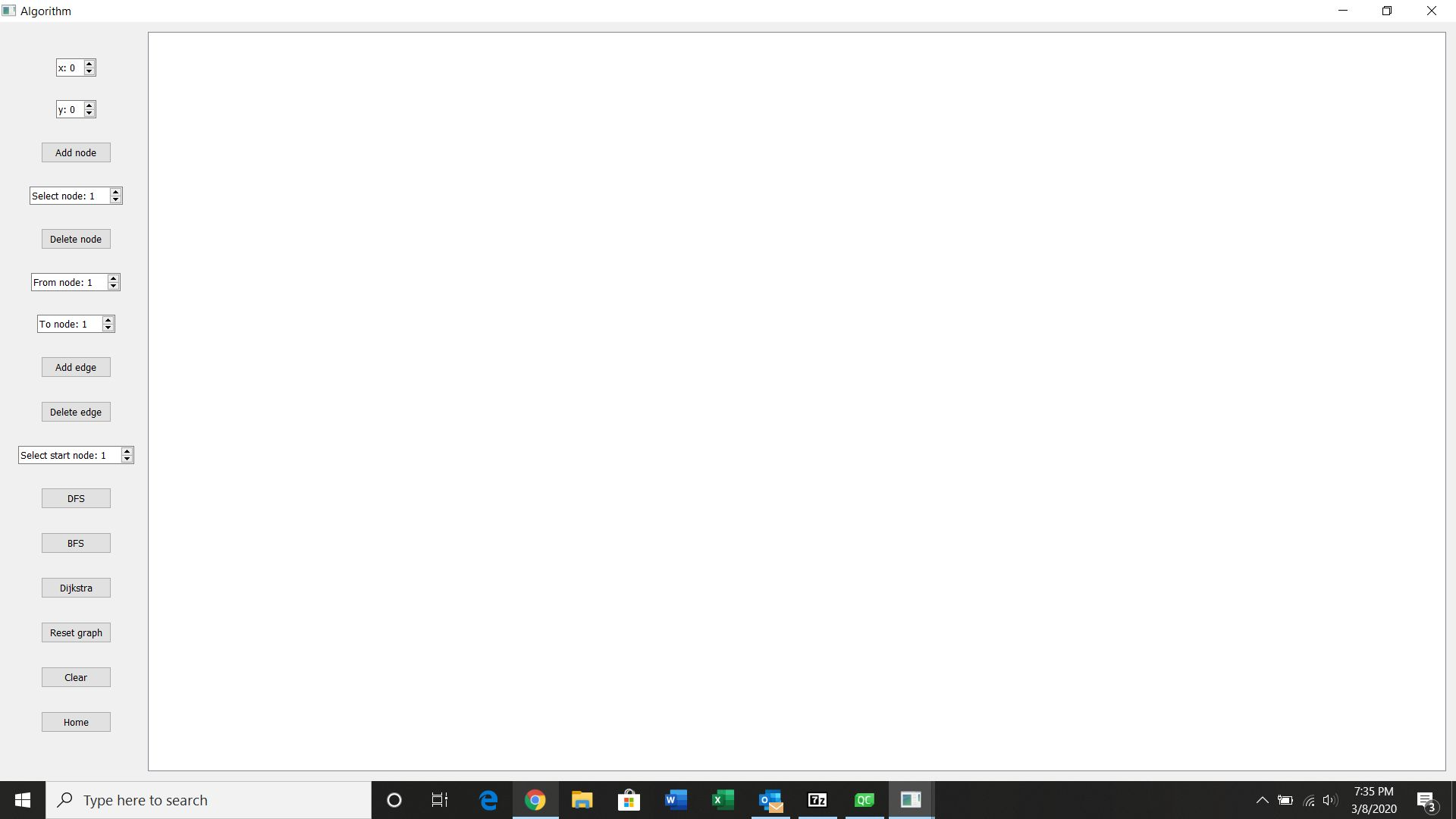
Bridges of Königsberg



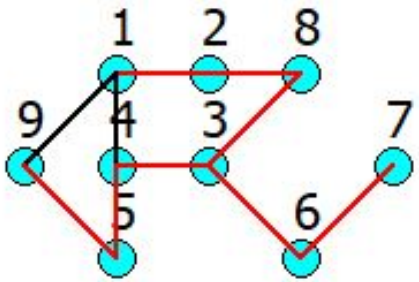
Graphs Class Overview

- Edge class: parameters of edge, e.g. source node & destination node.
- Node class: parameters of the node, e.g. neighbors, coordinate and node number

- GraphWindow class: contains all the buttons that allows addition/removal of nodes/edges, and signals algorithms when pressed.
- GraphWidget class: the graph that exists in GraphWindow; contains vectors of nodes and edges and paints them according to the user’s input


Algorithm
x: 0
y: 0
Add node
Select node: 1
Delete node
From node: 1
To node: 1
Add edge
Delete edge
Select start node: 1
DFS
BFS
Dijkstra
Reset graph
Clear
Home
Type here to search
7:35 PM
3/8/2020

EXAMPLE OF DFS (Starting at node 1)



BFS (starting at node 1)