# **Assignment 4 - Graphs**

This assignment is a Graph Search algorithm based on the **Generic Search** algorithm provided as a set of starter files on the Week 4 Canvas module for this course. The algorithm is described as a Reachability Problem in chapter 14.1 of the course text. It takes as input an undirected graph ***G*** and a source node ***s***, and returns a collection containing all the nodes that are reachable by a path in ***G*** from ***s***. See **chapter 14.1** of the course text for a thorough description of the algorithm, including pseudo code.

Chapter 14.2 of the course text presents a modification of **Generic Search** algorithm. A couple of changes to the Generic Search algorithm transform it into a **Search for Shortest Paths** algorithm. Given an undirected graph G and a source node s, the algorithm outputs a collection with the following properties for each node ***v***:

* the length of the shortest path from the source node ***s*** to **v**
* the shortest-path

The best place to start with this assignment is to make sure you read and understand chapter 14.1. The last section discusses the order that the nodes are handled in the Generic Search. The author suggests that changing the data structure of foundNotHandled from a set to a queue will  find for each node v a shortest path from s to v.

Because the data returned by the new algorithm contains new data, you will need to create a new data structure. One that includes for each node ***v***, the path length from **s** to ***v***, and the path from ***s*** to node ***v***.

Your algorithm should be implemented as a method named ***ShortestPathBreadthFirst*** in the***class Graph*** (provided as starter code). It should take two parameters:

* Graph ***g***
* Vertex***s***

and return the paths and corresponding distances of those paths to nodes that are reachable from ***s***.

When you are satisfied with your implementation, upload it to Canvas along with any other files you modified.