Joey Troyer

Vishwanathan Roopa

CS 372

20 April 2023

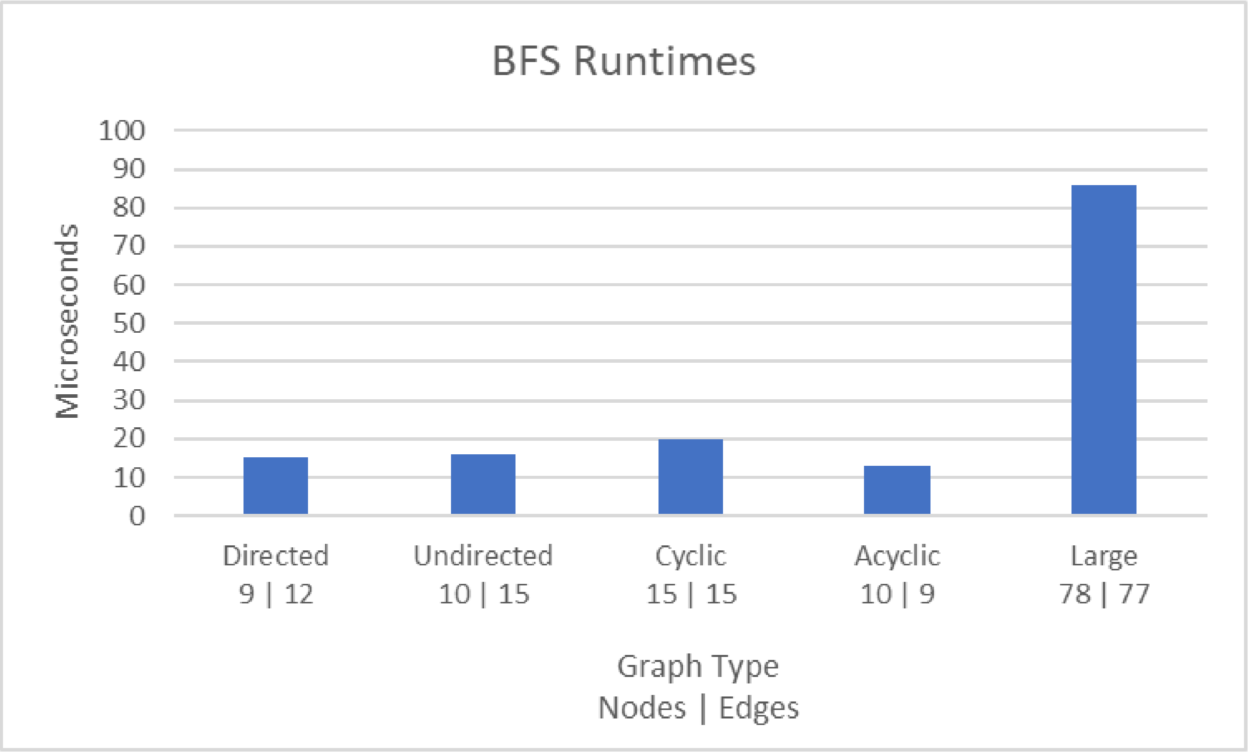
INTRODUCTION

In this lab, we extended our previous work on implementing a graph using an adjacency list and traversing it using a depth first search (DFS) algorithm by implementing the breadth first search (BFS) algorithm. The motivation behind this lab was to gain a deeper understanding of graph traversal techniques and learn how to apply BFS to solve problems involving shortest path graph traveral. To implement the BFS algorithm, we modified the Node class to include a distance attribute, which we used to store the distance from the source node to each node in the graph. We then implemented the BFS algorithm using a queue to get through all the discovered but not yet processed nodes. By tracking the distance from the source node to each node encountered, we were able to find the shortest path between the source node and all other nodes in the graph.

METHODS

The methods I used in implementing the breadth first search (BFS) algorithm. I updated the Node class, adding a distance attribute to store the shortest distance from the source node. The distance for all nodes is initialized to -1. Then, I implemented the BFS algorithm using a deque data structure to maintain a queue of discovered but not yet processed nodes. I would dequeue the front node update the distances of its adjacent nodes, and then added the adjacent nodes that haven't already been visited to the queue and loop until the queue is empty. This allowed me to find the shortest path to all reachable nodes.

RESULTS



The BFS algorithm I have implemented seems to be consistent with the runtime of O(|V | + |E|). We can say this because as the number of nodes (|V|) and edges (|E|) increase, the execution time also increases accordingly.

CONCLUSION

In this lab, we successfully extended our graph implementation and traversal methods by incorporating the breadth first search (BFS) algorithm. The BFS algorithm effectively found the shortest distances between the source node and all other nodes in the graph. Some issues with our implementation is that our BFS algorithm might not be the most efficient implementation and it could probably be improved more.