CS 261 Data Structures

Student: Joaquin Saldana

Answers Section of Assignment 7

1. **How is the graph stored in the provided code? Is it represented as an adjacency matrix or edge list?**

   *The graphs provided are represented as an adjacency matrix. This is so, since the output shows the number of vertices and the vertices in which it connects to or as some will call it its "neighbors"*

2. **Which of the 3 graphs are connected? How can you tell?**

   *Graph 2 is the only graph that is "connected". Each of the vertices has an edge to each of the other vertices.*

3. **Imagine that we ran each depth-first and breadth-first searches in the other direction (from destination to source). Would the output change at all? Would the output change if the graphs were directed graphs?**

   *If the graph was directed, then the output will change. This is because directed graphs/edges have only one direction either in DFS or BFS the route is unique.*

4. **What are some pros and cons of DFS and BFS? When would you use one over the other?**

   *As per the slides and video, there are several pros and cons for each. DFS is similar to a single person working a maze while BFS is as if multiple people are working on the maze at the same time.*

   *BFS may not find the path to the destination the quickest, but will ALWAYS find it eventually, and if the maze happens to have an infinite path BFS will not get stuck in it.*

   *Meanwhile, with DFS if there is an infinite path there is a chance it will get stuck in it. DFS however uses less memory than BFS algorithm and could potentially be much quicker than BFS.*

5. **What is the Big O execution time to determine if a vertex is reachable from another vertex?**

   *Per the lecture, both DFS and BFS execute at O(E) at most in the while loop section of code for both DFS and BFS. E standing for the number of edges in the graph.*