Sonya Carmona

Project 2 Documentation

1. Describe the data structure you used to implement the graph and why? [2.5 points]

The data structure I used to implement the graph was a map with the first value being the string name of the vertex(to) and the second value being a pair with a vector string for the vertex(from) that points to the first string value. The second value of the pair is a float value of one over the outdegrees of the first pair vertex. This formed and created the adjacency list because it favors sparse graphs for space efficiency. Adjacency lists are also easy to add and delete vertices.

2. What is the computational complexity of each method in your implementation in the worst case in terms of Big O notation? [5 points]

The computational complexity of each method in my implementation in the worst case in terms of Big O notation is in order as follows. My addVertex method is O(n) due to the number of vertices being added and checked if they are unique. My printAllVertex is the same with an O(n) complexity, with n being the number of vertices. addEdges is also O(n) because it searches and finds if the vertex exists. addAdjList has a complexity of O(m*n) because m is the number of edges and n is the number of vertices. outDegree has a complexity of O(n) because this method iterates through all the vertices. getOutDegree has a complexity of O(n) of n being the number of vertices to iterate through. inDegrees is O(m) because it iterates through the number of edges. My calculate PageRank algorithm is O(m*powerIterations), m is the number of edges and because the power iterations is a parameter it does that amount of times.

3. What is the computational complexity of your main method in your implementation in the worst case in terms of Big O notation? [5 points]

The computational complexity of the main method in my implementation in the worst case is O(numLines*(addVertex +addEdges) + addAdjList + outDegrees + inDegrees + (calculatePageRank*powerIterations)) because all the methods are called in the main.

4. What did you learn from this assignment and what would you do differently if you had to start over? [2.5 points]

What I learned from this assignment is how to implement adjacency lists and how to create them. What I could do differently is the format of my map to make it easier to multiply it by a vector.