

## Page Rank Report

For my project I decided to use a map to represent my adjacency list. The reason I did this is because in the background of c++, maps are represented as a red-black tree. This means that operations like search, insert, and delete have a time complexity of  $O(\log(n))$ , which in turn makes my adjacency list efficient.

## Computational Complexities

1. **insertVertex:** My insertVertex method has a computational complexity of  $O(|V|)$  because it is completely dependent on how many vertices a person wants to insert into the graph.
2. **pageRank:** My page rank method has a time complexity had a computational complexity of  $O(|V| * |E| * P)$  where P in this case represents the amount of power iterations required. My pageRank method has a nested for loop where the outer for loop checks all of the vertices and the nested for loop checks all of the edges. Hence, this is where the  $O(|V| * |E|)$  comes from. However, these for loops are within a while loop that is dependent on the amount of power iterations required, which is where  $O(|V| * |E| * P)$  comes from.
3. **Main Method:** My main method has a computational complexity of  $O(N + P)$  where n represents the number of lines to read and p represents the amount of time needed to run the pageRank() function. The number of lines serves as the condition for my while loop which is where the N comes from and the function call is not nested within the while loop which is why it would be + P.

## **Reflection**

From this assignment I learned how useful adjacency lists can be for storing data within c++. I can now take this knowledge and use it to implement different variations of graphs within different coding languages. I also learned the basics of how Google's page rank algorithm works at the surface level. I know what we did is not as complex as Google's algorithm, but is cool to see how the websites get ranked. If I were to start over I would have used an inDegree and an outDegree map sooner. These maps allowed me to simplify all of my calculations which eventually led me to passing all of the test cases.