Matthew F. Leader
July 26, 2017
Project 4 – Hash Tables and Spell Checking
CSC 316-651
Data Structures & Algorithms

SpellChecker Report

For my SpellChecker project, I decided to use Separate Chaining to resolve collisions. Effectively, this requires a HashTable that has an array of linked lists of hash nodes. Each hash node stores an individual key, its associated value, and the next hash node in the list.

To create my hash function, I paired the polynomial hash function that was discussed in class to hash String objects with the golden ratio compression method to form a complete hash function for my HashTable class.