CSci 115 - Spring 2021
Algorithms and Data Structures
Lab: Hash Table

Introduction:

This lab's focus is on hashing. Hashing is a useful technique that allows a programmer to retrieve/insert an element from anywhere in an array in near instant time (O(1)). It does this by mapping a hash function hash() from some domain (our lab will use previous power-ball numbers) to positions in the array. Hashing is also used in a <key,value> pair where we can use a hash key to find a value. In a perfect hash function, each key would hash to a unique position in the array.

The Project:

On canvas there is a file called numbers.txt with the past 100 power-ball numbers and HashTable.h. Your goal is to create a program to take the sum of each days power-ball numbers and hash it into an array of size 50. You can simply use todaysSum % 50 as a hashing function. If two days have the exact same sum increase a counter. If two days hash to the same value but the sum is different use separate chaining to put the newest sum at the end of a linked list.

Your program should have a display function that neatly outputs each unique sum and how often each sum occurred.

Steps:

Create a Node Class that can hold an integer sum and an integer count (for how many times a sum occurred).

Create array of Node* of size 50 full of nullptrs.

For every day
sum up the day's 7 numbers
hash the sum using the hash function.
Search the nodes at position hash(sum)
If you find a node already exists with the same sum increase that nodes count, Otherwise create a new node for the sum with count=1
Make sure to put the logic for adding a new sum in HashTable.cpp
Display all the hashed sums, where they are hashed to, and their respective count.

Requirements:

1. Your program must compile
2. Use comments and meaningful names for the variables
3. Use good object oriented design (Separate the declaration and implementation using a provided header file)
4. Cleanly display the correct values. (it will allow you to verify that it works properly)