**Lab 7**

**Hash Table**

**Description**

Using the STL vector and list classes, create a Hash Table ADT. The data in the hash table must be stored in a HashNode class that contains both the key and the value. The HashTable class must have at least the following functions besides the canonical functions:

void Insert(K key, V value);

void setHash(int (\*hash)(K key)); // store the function ptr in data member

V operator [] (K key);

void Delete(K key);

void Traverse(void (\*visit)(V value));

Be sure to “rehash” the table if the function pointer changes. Test the table with the following data type.

struct Book

{

string m\_title;

string m\_author;

int m\_pages;

};

Use the ISBN (string) as the key for the data. Therefore, you should be able to instantiate an object and test with something like the following.

HashTable<string, Book> table(10);

table.setHash(AsciiHash);

Book temp = {"C++: An Active Learning Approach", "Randal Albert", 635};

table.Insert("0763757233", temp);

Book temp1 = {"Rodeo for Dummies", "Calvin Caldwell", 1};

table.Insert("7063757233", temp1);

Book temp3 = {"And That n There", "Ralph Carestia", 1};

table.Insert("7063757234", temp3);

cout << table["0763757233"].m\_title << endl;

cout << table["7063757233"].m\_title << endl;

cout << table["7063757234"].m\_title << endl;