//header file for the hash table, aka address book

#include "Client Info BST.h"

class Client\_Address\_Book{

public:

Client\_Address\_Book(); //default constructor will read data from input file "client\_address\_data.txt".

Client\_Address\_Book(const Client\_Address\_Book &); //Copy Constructor

~Client\_Address\_Book(); //Destructor

void Insert(const string &); //Insert adds a new Client's information to the hash table

void Remove(const string &); //remove record

//Remove deletes a client from the hash table if it is there; otherwise a message should be printed stating so.

void Update(const string &); //update record; see example below

void Print\_BST(const string &); //Print a BST (cell in hash table) inorder to the screen

void Print\_Hash\_Table(); //function will print hash table to the screen

void Print\_Hash\_Table\_to\_File(const string & filename); //function will print hash table to output file

bool \* Search(const string &); //return true if client found; otherwise false

unsigned int Hash\_Function(const string &); //return the index of the BST in the hash table

//Hint:Remember that the insert, remove and search function for Clients\_Address\_Book will use Client\_Info\_BST’s insert, remove and search respectively.

private:

int capacity; //SET THIS VALUE EQUAL TO 27 YOUR DEFAULT CONSTRUCTOR //STOP CAPS LOCKING

Clients\_Info\_BST \* hash\_table; // USING 1 THROUGH 26 or whatever you like

};