#include <iostream>

#include <string>

#define PRIME\_SIZE 120

using namespace std;

class NewClass

{

public:

NewClass\* next;

int val;

NewClass()

{

this->next = NULL;

}

NewClass(int val)

{

this->val = val;

this->next = NULL;

}

~NewClass()

{

if (this->next != NULL)

{

delete this->next;

}

}

};

class HashTable

{

NewClass\* table[PRIME\_SIZE];

static int hash(int Value)

{

return Value % PRIME\_SIZE;

}

public:

HashTable()

{

for (int i = 0; i < PRIME\_SIZE; i++)

{

table[i] = NULL;

}

}

~HashTable()

{

for (int i = 0; i < PRIME\_SIZE; i++)

{

delete table[i];

}

}

NewClass\* find(int val)

{

int hashNumber = hash(val);

NewClass\* result = table[hashNumber];

if (!result)

{

cout << " Елемент не знайдено" << endl;

cout << endl;

return NULL;

}

while (result->val != val)

{

if (!result->next)

{

cout << " Елемент не знайдено" << endl;

cout << endl;

break;

}

result = result->next;

}

return result;

}

void deleteElem(int val2)

{

int hashNumber = hash(val2);

NewClass\* result = table[hashNumber];

if (!result)

{

cout << " Елемент не знайдено" << endl;

cout << endl;

}

else {

if (result->next == NULL)

{

cout << " Видалено елемент " << result->val << endl;

cout << endl;

result->val = 0;

result = result->next;

table[hashNumber] = NULL;

delete result;

}

else {

if (result->val == val2)

{

NewClass\* result4 = result;

result4->next = result->next;

result4 = result4->next;

table[hashNumber] = result4;

}

NewClass\* result3 = result;

NewClass\* result2 = result;

int mar33 = 1;

while (result3->val != val2)

{

if (result3->next == 0)

{

cout << " Елемент не знайдено" << endl;

cout << endl;

mar33 = 0;

break;

}

//else if (result3->age == age2) {

// mar33 = 1;

//}

else {

result2 = result3;

result3 = result3->next;

}

}

if (mar33 == 1) {

cout << " Видалено елемент " << result3->val << endl;

cout << endl;

result2->next = result3->next;

}

}

}

}

void push(int val)

{

int hashNumber = hash(val);

NewClass\* pers = new NewClass(val);

NewClass\* place = table[hashNumber];

if (place == NULL)

{

table[hashNumber] = pers;

return;

}

while (place->next != NULL)

{

place = place->next;

}

place->next = pers;

}

int numElem(int it)

{

NewClass\* result = table[it];

if (!result)

{

int result2 = 0;

return result2;

}

else {

int result2 = result->val;

return result2;

}

}

// перевірка на порожність

int perevNulls(int it2) // 0 - елемента немає

{

NewClass\* result5 = table[it2];

int markperev;

if (!result5)

{

markperev = 0;

}

else { markperev = 1; }

return markperev;

}

void vivod2(int it)

{

NewClass\* result4 = table[it];

if (!result4) { cout << " "; }

else {

if (!result4->next)

{

cout << " ";

}

else {

while (result4->next != NULL)

{

result4 = result4->next;

int newNum2 = result4->val;

cout << " --> " << newNum2;

}

}

}

}

};

int main()

{

HashTable newTable;

int numbers;

string name1;

string surname1;

int val1;

int val2;

string name2;

int zmdelElem;

int randElem;

do

{

cout << endl;

cout << " 1. Додати елемент " << endl;

cout << " 2. Пошук " << endl;

cout << " 3. Вивести всі елементи " << endl;

cout << " 4. Видалити елемент " << endl;

cout << " 0. Вийти\n\n";

cout << " Номер команди > "; cin >> numbers;

cout << endl;

switch (numbers)

{

case 1:

{

cout << " Введіть число > "; cin >> val1; cout << endl;

newTable.push(val1);

cout << " Додано елемент" << endl;

cout << " " << val1 << endl;

break;}

case 2:

{

cout << " Введіть шукане число > "; cin >> val2;

cout << endl;

NewClass\* search = newTable.find(val2);

if (search)

{

cout << " Елемент " << search->val << " в таблиці існує " << endl;

cout << endl;

}

}

break;

case 3:

{ for (int i = 0; i <= PRIME\_SIZE - 1; i++)

{

int newmarPer = newTable.perevNulls(i);

if (newmarPer == 0)

{

if (i == PRIME\_SIZE - 1) {

for (int i = 0; i <= PRIME\_SIZE; i++) {

if (i < PRIME\_SIZE) {

int newmarPer = newTable.perevNulls(i);

if (newmarPer == 0)

{

continue;

}

else { break; }

}

else { cout << " Таблиця поки що пуста " << endl; }

}

}

else { continue; }

}

else {

int result3 = newTable.numElem(i);

cout << " KEY = " << i << " elem = " << result3;

newTable.vivod2(i);

cout << endl;

}

}

cout << endl;

}

break;

case 4:

{

cout << " Який елемент видалити? > "; cin >> zmdelElem;

cout << endl;

newTable.deleteElem(zmdelElem);

}

break;

case 0: {break;}

default: { cout << endl << " Команда не визначена\n\n";

break;}

}

} while (numbers != 0);

//system("pause");

return 0;

}