Министерство образования Республики Беларусь

Учреждение образования

«Белорусский государственный университет информатики

и радиоэлектроники»

Специальность «Программная инженерия»

Кафедра инженерной психологии и эргономики

Учебная дисциплина «Основы алгоритмизации и программирования»

ОТЧЕТ

по лабораторной работе №15

«Хеширование»

Вариант 27

Подготовил: Сивцов А.Д.

Проверил: Усенко Ф.В.

Минск 2024

**Цель работы:** сформировать знания и умения по работе с подпрограммами, приобрести навыки написания программ с использованием хеш-функций.

Задание: Составить хеш-функцию в соответствии с заданным вариантом и проанализировать ее. При необходимости доработать хеш-функцию. Используя полученную хеш-функцию разработать на языке программирования С++ программу, которая должна выполнять следующие функции:

* создавать хеш-таблицу;
* добавлять элементы в хеш-таблицу;
* просматривать хеш-таблицу;
* искать элементы в хеш-таблице;
* удалять элементы из хеш-таблицы.

#include <iostream>

#include <string>

#include <list>

using namespace std;

struct hashTable {

list<string>\* table = nullptr;

int capacity;

hashTable() {}

hashTable(int size) : capacity(size) {

table = new list<string>[capacity];

}

int hashFunction(string key) {

int hash = 0;

for (char c : key) {

hash += static\_cast<int>(c);

}

hash = hash \* hash;

hash = hash / 100;

hash = deleteFirstTwoNums(hash);

while (hash > 2000) {

hash = hash % 2000;

}

return hash % capacity;

}

void insert(string key) {

int index = hashFunction(key);

table[index].push\_back(key);

}

void display() {

for (int i = 0; i < capacity; i++) {

cout << "Index " << i << ": ";

for (string& value : table[i]) {

cout << value << " ";

}

cout << endl;

}

}

void search(string key) {

int index = hashFunction(key);

for (string value : table[index]) {

if (value == key) {

cout << "Элемент " << key << " найден на позиции " << index << endl;

}

}

}

void remove(string key) {

int index = hashFunction(key);

table[index].remove(key);

}

int deleteFirstTwoNums(int a) {

int p = 1;

while (a / p > 99) p \*= 10;

int y = a % p;

return y;

}

};

int main() {

setlocale(LC\_ALL, "Russian");

int cmd;

hashTable\* container = nullptr;

string key;

do {

cout << "1 - создать хеш-таблицу" << endl;

cout << "2 - добавить элемент" << endl;

cout << "3 - просмотреть хеш-таблицу" << endl;

cout << "4 - поиск элемент" << endl;

cout << "5 - удалить элемент" << endl;

cout << "0 - выход" << endl;

cout << "Введите команду: " << endl;

cin >> cmd;

switch (cmd) {

case 0: {

break;

}

case 1: {

cout << "Введите размер таблицы: " << endl;

int size;

cin >> size;

container = new hashTable(size);

cout << "Хеш-таблица успешно создана!" << endl;

break;

}

case 2: {

cout << "Введите строку: " << endl;

cin.ignore();

getline(cin, key);

container->insert(key);

break;

}

case 3: {

container->display();

break;

}

case 4: {

cout << "Введите строку: " << endl;

cin.ignore();

getline(cin, key);

container->search(key);

break;

}

case 5: {

cout << "Введите строку: " << endl;

cin.ignore();

getline(cin, key);

container->remove(key);

break;

}

default: {

cout << "Неизвестная команда! Попробуйте снова." << endl;

break;

}

}

} while (cmd != 0);

delete[] container;

return 0;

}

Результат работы программы представлен на рисунке 1

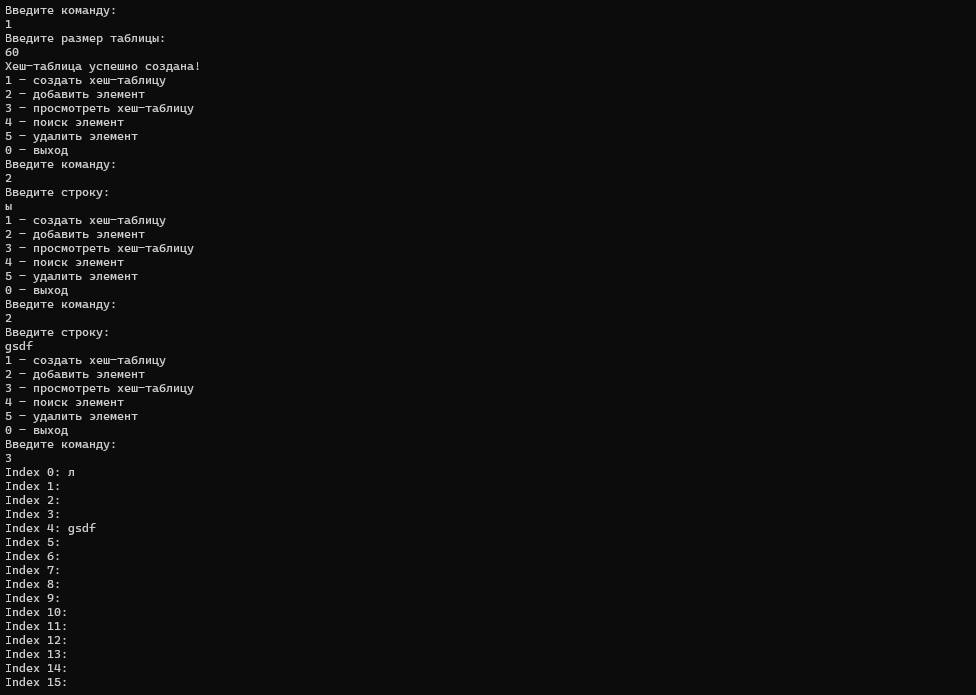


Рисунок 1 – Результат работы программы

Блок-схема работы программы представлена на рисунке 2.

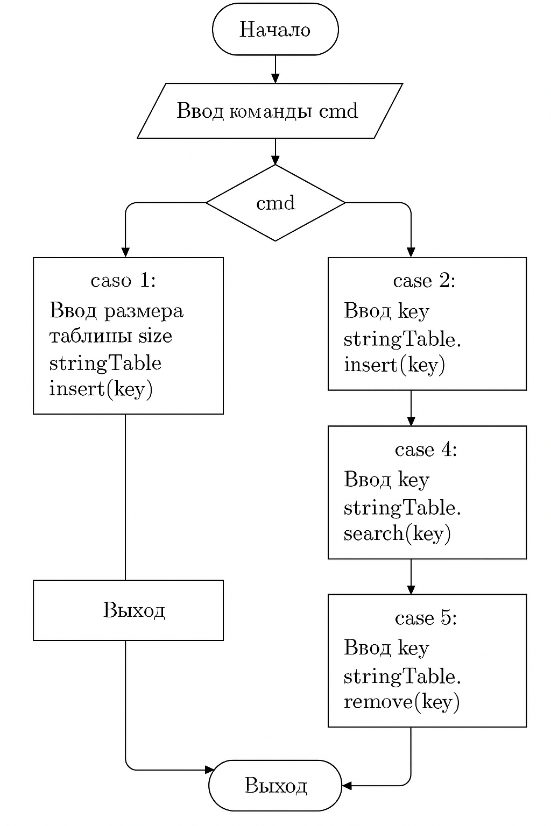


Рисунок 2

**Вывод:** в ходе выполнения работы была достигнута цель данной лабораторной работы: сформировать знания и умения по работе с подпрограммами, приобрести навыки написания программ с использованием хеш-функций.