**Пермский национальный исследовательский политехнический университет**

Кафедра “Информационные технологии и автоматизированные системы”

**Лабораторная работа №18**

По дисциплине «Основы алгоритмизации и программирования»

**Тема:**

Хеш-таблицы.

Выполнила:

студентка группы ИВТ-19-1б

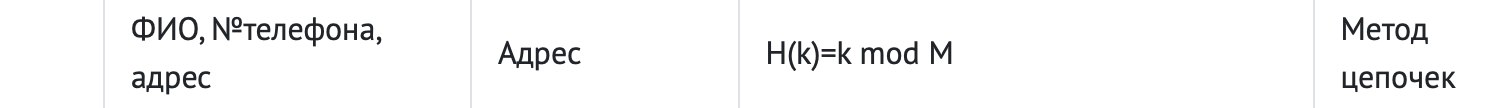
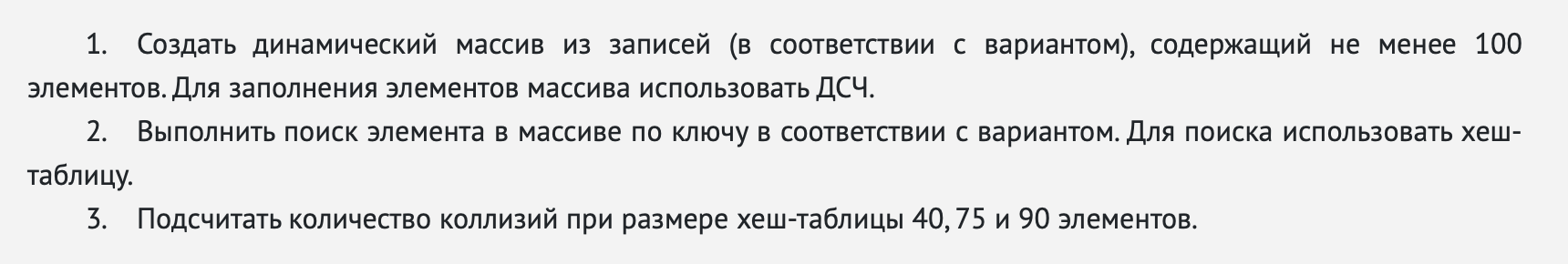
Еремеева Полина Алексеевна

Проверила:

доцент кафедры “ИТАС”

Полякова О.А.

Пермь, 2020

**Постановка задачи**

**Код**

#include <iostream>

#include <string>

#include <ctime>

#include <math.h>

**using** **namespace** std;

**const** **int** size40 = 40, size75 = 75, size90 = 90;

**struct** person {

string name;

string adress;

**unsigned** **long** **int** tel\_num = 0;

person\* next = **NULL**;

};

string mid[] = { "Иванов","Петров","Сидоров","Тесла","Маск","Эйнштейн","Ньютон","Гук","Кюри","Сталин","Ленин","Маркс" };

string first[] = { "Иван","Никита","Егор","Коля","Вика","Альберт","Толя","Роберт","Лера","Мария","Дмитрий","Карл","Алексей","Михаил","Владимир" };

string last[] = { "Иванович","Петрович","Сидорович","Николаевич","Илонович","Альбертович","Исаакович","Робертович","Маркович","Иосифович","Владимирович","Карлович","Алексеевич","Михаилович","Дмитриевич" };

string dobs[] = { "Павлова 54", "Ленина 59", "Мира 56", "Плеханова 76","Кирова 50","Целая 43","Калинана 65","Калугина 34","Гамашова 53" };

**struct** chains {

person\* first = 0;

**int** size = 0;

**void** push(person\* sled) {

person\* p = first;

**if** (p) {

**while** (p->next)p = p->next;

p->next = sled;

}

**else** {

first = sled;

}

size++;

}

**bool** empty() { **return** !**this**->size; }

};

**int** new\_key(string data, **int** hash) {

**int** key = 0;

**for** (**int** i = 0; i < data.size(); i++) {

key = key + ((**int**)(data[i]) \* i \* 9);

}

**return** (hash \* ((key \* 0.61) - trunc(key \* 0.61)));

}

**void** add\_el(person\* news, **int** hsize, chains\* news\_massiv\_hash) {

**int** i;

i = new\_key(news->adress, hsize);

news\_massiv\_hash[i].push(news);

}

**void** find(**int** i, string data, chains\* mass) {

**if** (mass[i].empty()) {

cout << "Такого человека нет!" << endl;

}

**else** {

**if** (mass[i].size == 1) {

**if** (mass[i].first->adress == data) {

cout << "ФИО:\t" << mass[i].first->name;

cout << endl << "Адрес: " << mass[i].first->adress;

cout << endl << "Номер телефона: +7" << mass[i].first->tel\_num;

**return**;

}

**else** {

cout << endl << "Такого человека нет!";

}

}

**else** {

person\* f = mass[i].first;

**while** (f) {

**if** (f->adress == data) {

cout << "ФИО:\t" << mass[i].first->name;

cout << endl << "Дата рождения: " << mass[i].first->adress;

cout << endl << "Номер телефона: +7" << mass[i].first->tel\_num;

**return**;

}

**else** {

f = f->next;

}

}

cout << "Такого человека нет!" << endl;

}

}

}

**void** search(**int** hsize, chains\* news\_massiv\_hash) {

string search\_adress;

cin.ignore(32767, '\n');

cout << "Введите адрес: ";

getline(cin, search\_adress, '\n');

**int** i = new\_key(search\_adress, hsize);

find(i, search\_adress, news\_massiv\_hash);

}

**void** collisions(**int** hsize, chains\* news\_massiv\_hash) {

**int** k = 0;

**for** (**int** i = 0; i < size40; i++) {

**if** (news\_massiv\_hash[i].size) {

**if** (news\_massiv\_hash[i].size > 1) {

k++;

}

}

}

cout << "\nКоличество коллизий: " << k << '\n';

}

**int** main()

{

setlocale(LC\_ALL, "rus");

srand(time(0));

**int** n;

cout << "Введите размер: ";

cin >> n;

chains\* news\_massiv\_hash = **new** chains[n];

**for** (**int** i = 0; i < n; i++)

{

person\* news = **new** person;

news->name = mid[rand() % 12] + ' ' + first[rand() % 15] + ' ' + last[rand() % 15];

news->adress = dobs[rand() % 9];

news->tel\_num = 10000000000 + rand() % 99999999999;

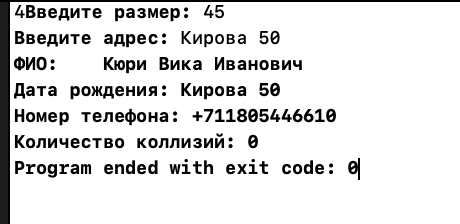
add\_el(news, n, news\_massiv\_hash);

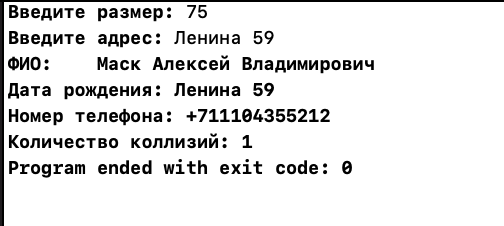
}

search(n, news\_massiv\_hash);

collisions(n, news\_massiv\_hash);

}

**Тестирование**

****

