Министерство образования и науки Российской Федерации  
Федеральное‌ ‌государственное‌ ‌бюджетное‌ ‌образовательное‌ ‌учреждение‌

высшего‌ ‌образования‌

**«Пермский национальный исследовательский  
политехнический университет»**

Кафедра «Информационные технологии и автоматизированные системы»

**ОТЧЁТ**

**по лабораторной работе №15**

Дисциплина: «Основы алгоритмизации и программирования»

Тема: Методы внутренней сортировки массивов: быстрая сортировка, сортировка подсчётом

Вариант 1

Выполнил:

Студент группы РИС-20-1б

Азмагулов Артём Вадимович

Проверила:

Доцент кафедры ИТАС

Полякова О. А.

**Пермь, 2021**

**Цель работы**

Ознакомление с методами внутренними сортировками массивов

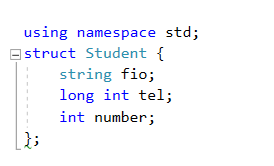
**Постановка задачи**

Реализовать внешние сортировки Шелла и Хоара на С++

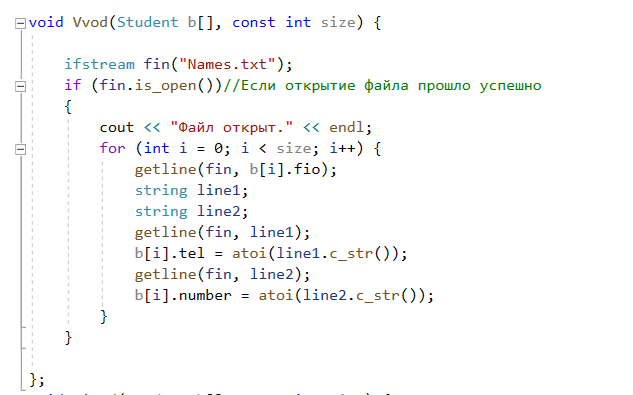
**Анализ задачи**

**1.** Для решения задачи необходимо:

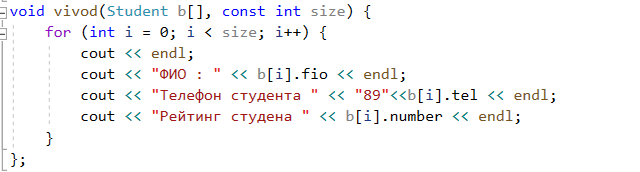
**1.1.** Реализовать структуру для дальнейшей работы с быстрыми сортировками



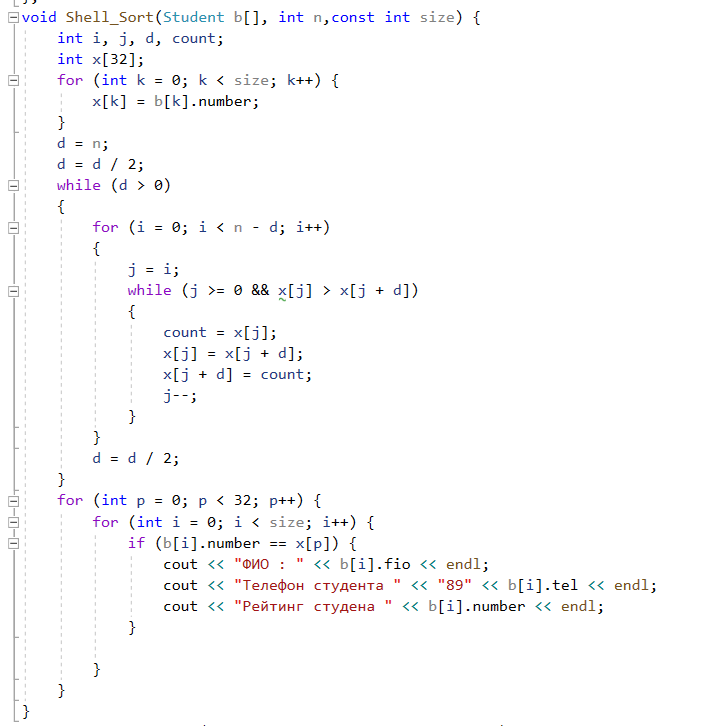
**1.2.** Реализовать функцию для приема данных в файл:



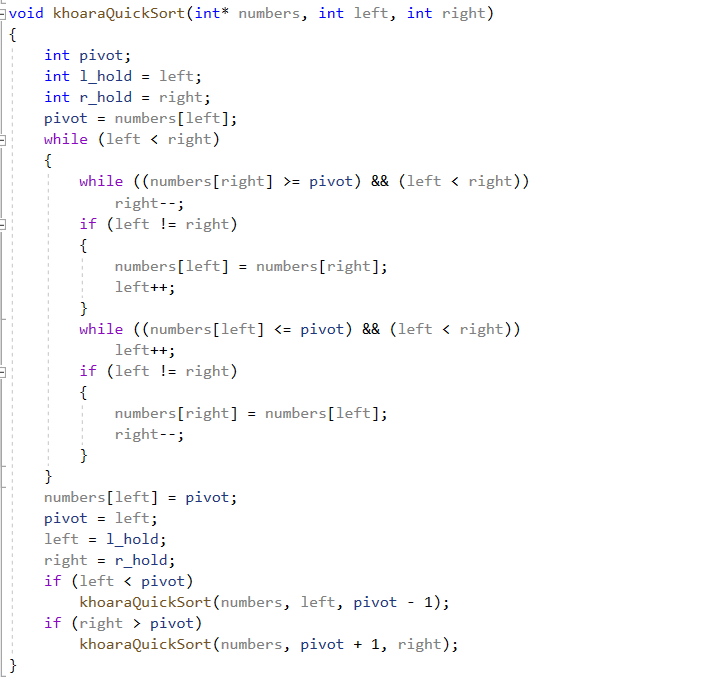
**1.3.** Реализовать функцию для вывода данных на консоль:



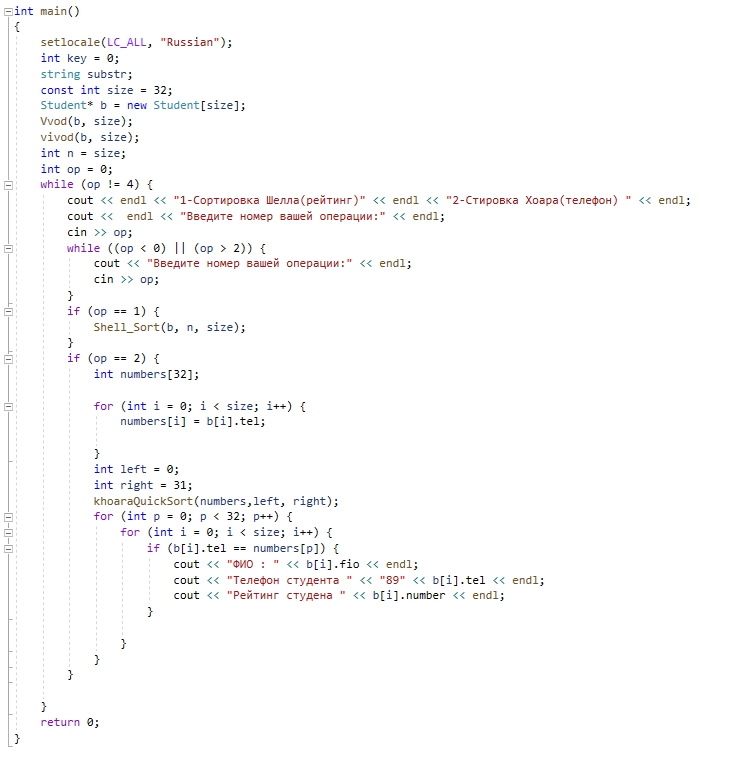
**1.4.** Реализовать функцию для демонстрации внутренней сортировки Шелла:



**1.5.** Реализовать функцию для демонстрации внутренней сортировки Хоара:

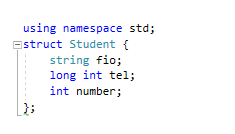


**1.6.** Реализовать меню в главной функции, с возможностью выхода из программы:



**2.**В ходе работы были использованы следующие типы данных:

**2.1.** Для составления структуры данных использовались данные string, int, long int.



**2.2.** Для работы с внутренней сортировкой Шелла использовались целочисленные данные:



**3.**Для решения задачи данные были представлены в следующем виде:

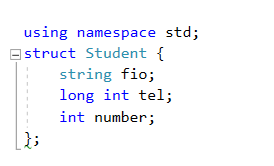
**3.1.** Для удобной обработки данных для элементов использовался класс int.

**4.**Для операций ввода и вывода использовались следующие операторы и функции:

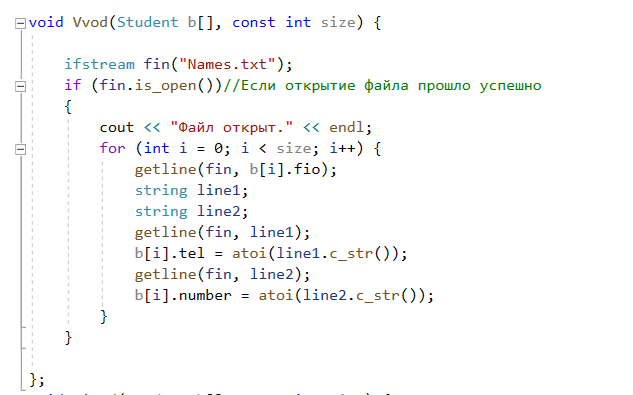
**4.1.**Вывод данных из массива реализован через функцию cout:



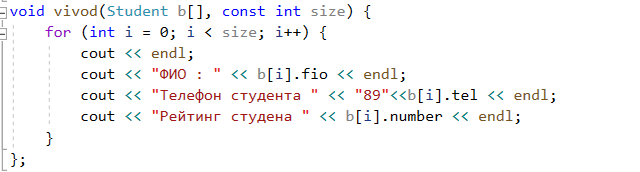
**5.**Поставленные задачи будут решены следующими действиями:

**5.1.** Для удобной обработки элементов было принято решение присвоить элементам структуры типы данных string,long int,int. 

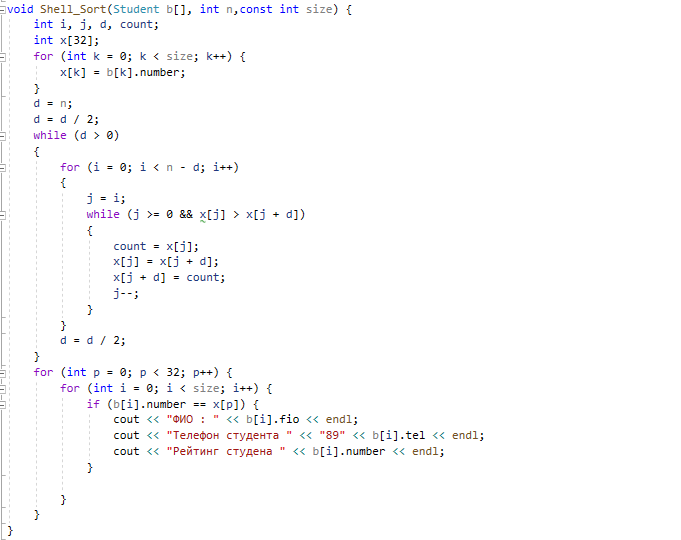
**5.2.** Для реализации ввода переменных в файл было принято решение по созданию файла Names.txt, где при помощи цикла for и функции getline заносятся данные в заданный файл:



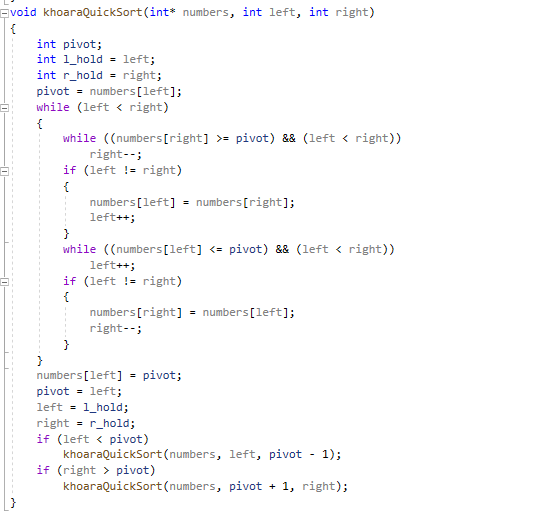
**5.3.** Для реализации вывода переменных используется цикл for, выводящий все элементы структуры через функцию cout:



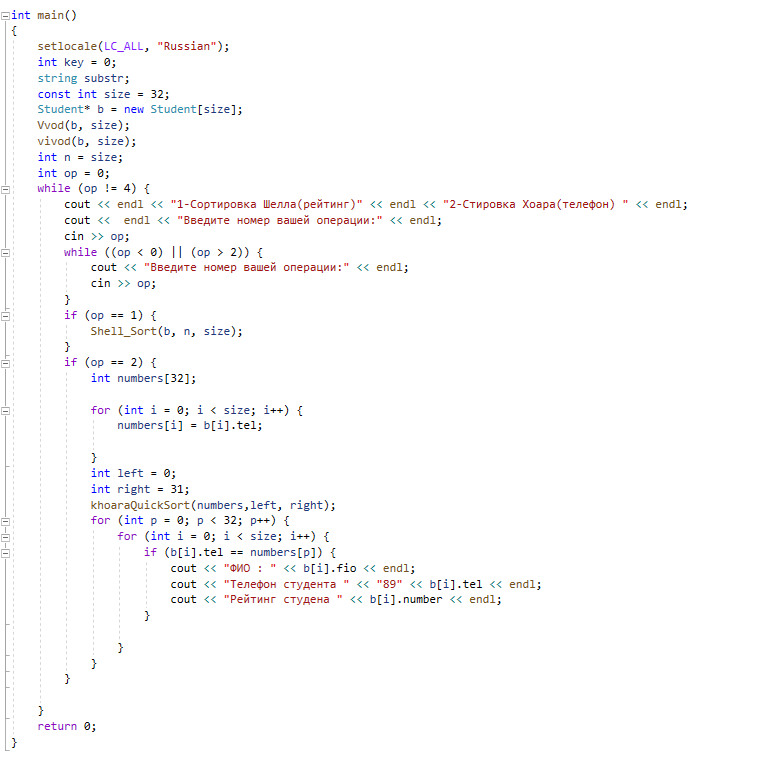
**5.4.** Для реализации сортировки Шелла используется цикл while сортирующий элементы до снижения собственного значения до 0, внутри данного цикла используется цикл for, меняющий данные массива в зависимости от сравнения его с предыдущим элементом.



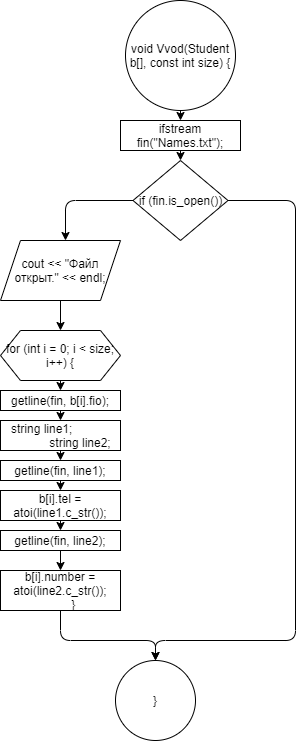
**5.5.** Для реализации сортировки Хоара используется цикл while сортирующий элементы до тех пор пока правая часть массива больше левой с дальнейшим использованием сортировки на примере рекурсии:

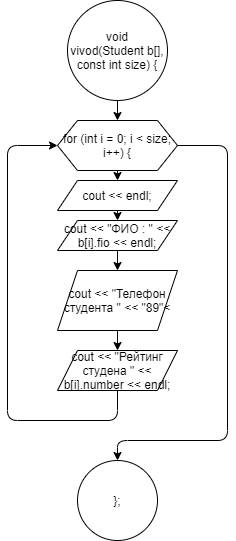


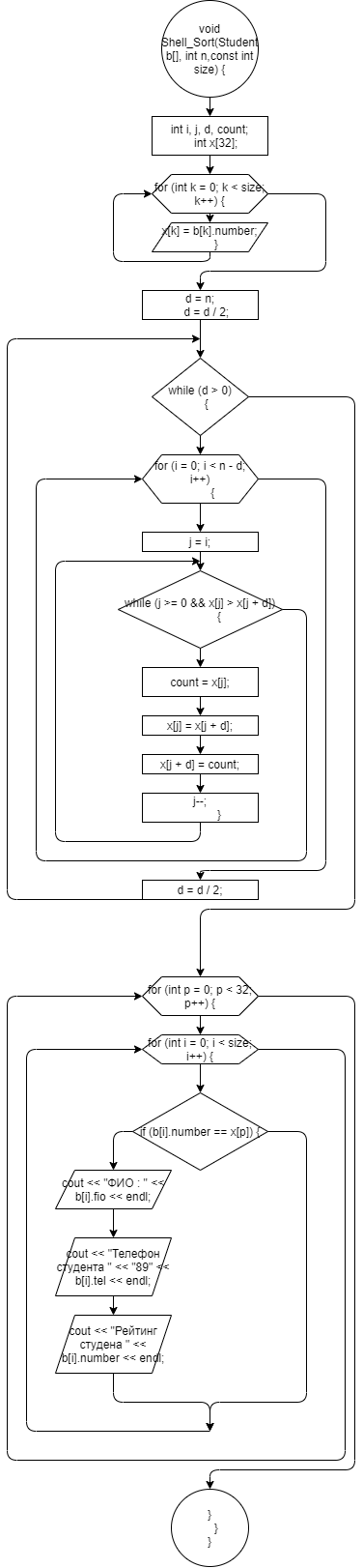
**5.6.** Для реализации меню было принято решение по созданию целочисленного элемента op для принятия значения со стороны пользователя для дальнейшей работы с нужным ему операцией:

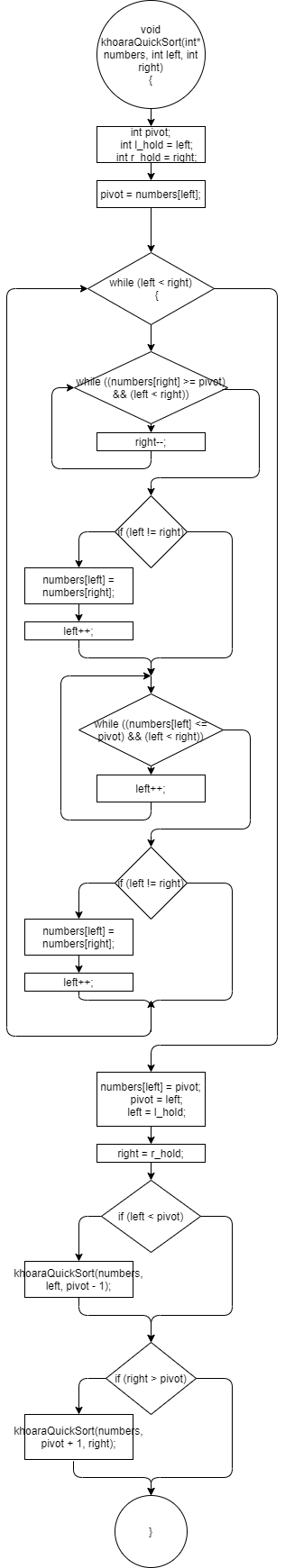


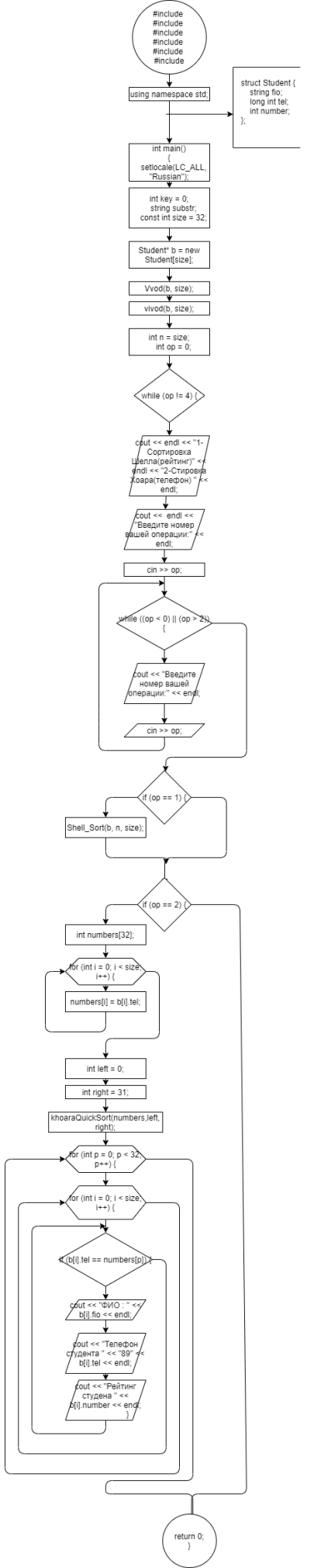
**Блок-схема программы**

****

****

****

****

****

**Решение**

#include <iostream>

#include <stdio.h>

#include <string>

#include <fstream>

#include <iostream>

#include <Windows.h>

using namespace std;

struct Student {

string fio;

long int tel;

int number;

};

void Vvod(Student b[], const int size) {

ifstream fin("Names.txt");

if (fin.is\_open())//Если открытие файла прошло успешно

{

cout << "Файл открыт." << endl;

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

getline(fin, b[i].fio);

string line1;

string line2;

getline(fin, line1);

b[i].tel = atoi(line1.c\_str());

getline(fin, line2);

b[i].number = atoi(line2.c\_str());

}

}

};

void vivod(Student b[], const int size) {

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

cout << endl;

cout << "ФИО : " << b[i].fio << endl;

cout << "Телефон студента " << "89"<<b[i].tel << endl;

cout << "Рейтинг студена " << b[i].number << endl;

}

};

void Shell\_Sort(Student b[], int n,const int size) {

int i, j, d, count;

int x[32];

for (int k = 0; k < size; k++) {

x[k] = b[k].number;

}

d = n;

d = d / 2;

while (d > 0)

{

for (i = 0; i < n - d; i++)

{

j = i;

while (j >= 0 && x[j] > x[j + d])

{

count = x[j];

x[j] = x[j + d];

x[j + d] = count;

j--;

}

}

d = d / 2;

}

for (int p = 0; p < 32; p++) {

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

if (b[i].number == x[p]) {

cout << "ФИО : " << b[i].fio << endl;

cout << "Телефон студента " << "89" << b[i].tel << endl;

cout << "Рейтинг студена " << b[i].number << endl;

}

}

}

}

void khoaraQuickSort(int\* numbers, int left, int right)

{

int pivot;

int l\_hold = left;

int r\_hold = right;

pivot = numbers[left];

while (left < right)

{

while ((numbers[right] >= pivot) && (left < right))

right--;

if (left != right)

{

numbers[left] = numbers[right];

left++;

}

while ((numbers[left] <= pivot) && (left < right))

left++;

if (left != right)

{

numbers[right] = numbers[left];

right--;

}

}

numbers[left] = pivot;

pivot = left;

left = l\_hold;

right = r\_hold;

if (left < pivot)

khoaraQuickSort(numbers, left, pivot - 1);

if (right > pivot)

khoaraQuickSort(numbers, pivot + 1, right);

}

int main()

{

setlocale(LC\_ALL, "Russian");

int key = 0;

string substr;

const int size = 32;

Student\* b = new Student[size];

Vvod(b, size);

vivod(b, size);

int n = size;

int op = 0;

while (op != 4) {

cout << endl << "1-Сортировка Шелла(рейтинг)" << endl << "2-Cтировка Хоара(телефон) " << endl;

cout << endl << "Введите номер вашей операции:" << endl;

cin >> op;

while ((op < 0) || (op > 2)) {

cout << "Введите номер вашей операции:" << endl;

cin >> op;

}

if (op == 1) {

Shell\_Sort(b, n, size);

}

if (op == 2) {

int numbers[32];

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

numbers[i] = b[i].tel;

}

int left = 0;

int right = 31;

khoaraQuickSort(numbers,left, right);

for (int p = 0; p < 32; p++) {

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

if (b[i].tel == numbers[p]) {

cout << "ФИО : " << b[i].fio << endl;

cout << "Телефон студента " << "89" << b[i].tel << endl;

cout << "Рейтинг студена " << b[i].number << endl;

}

}

}

}

}

return 0;

}

**Скриншоты**

