Министерство образования Республики Беларусь

г. Минск

Государственное учреждение образования

«Белорусский государственный университет информатики и радиоэлектроники»

Специальность «Инженерно-психологическое

обеспечение информационных технологий»

Учебная дисциплина «Основы алгоритмизации и программирования»

Отчет

по лабораторной работе №7

«Структуры и файлы»

Подготовил: Студент гр. 410901

Квитченко А.В

Проверил:

Усенко Ф.В

Минск 2024

***Цель работы:*** сформировать навыки и умения обработки структурированных типов данных, организованных в виде строковых данных

***Индивидуальное задание №18*** Создать структуру должностей Vacancies {Manager, Boss, Clerk, Salesman, etc.}. Создать структуру «Employee», состоящую из:

- поля name строкового типа;

- поля vacancy типа Vacancies;

- поля зарплата целого типа;

- поля дата приема на работу типа int[3].

Создать массив сотрудников. Длина массива задается пользователем, заполнение массива производится им же. Вывести полную информацию обо всех сотрудниках. Предусмотреть запись в файл

***Код программы приведен ниже:***

#include <iostream>

#include <string>

#include <fstream>

using namespace std;

enum Vacancies

{

Manager,

Boss,

Clerk,

Salesman,

};

struct Employee

{

string name;

Vacancies vacancy;

int salary;

int hireDate[3];

};

string Vacanciesstring(Vacancies vacancy)

{

switch (vacancy)

{

case Manager:

return "Manager";

case Boss:

return "Boss";

case Clerk:

return "Clerk";

case Salesman:

return "Salesman";

default:return "Unknown";

}

}

ostream& operator<<(ostream& os, const Employee& employees)

{

os << employees.name << endl << Vacanciesstring(employees.vacancy) << endl << employees.salary << endl << employees.hireDate[0] << "/" << employees.hireDate[1] << "/" << employees.hireDate[2] << endl;

return os;

}

int main()

{

setlocale(LC\_ALL, "Russian");

while (true) {

int Numbercontorls;

cout << "Введите 0 для выхода, введите 1 для записи сотрудников, введите 2 для просмотра списка сотрудников : ";

cin >> Numbercontorls;

fstream file;

file.open("Myfile.txt", fstream::in | fstream::out | fstream::app);

if (!file.is\_open())

{

cout << "Файл не удалось открыть!" << endl;

break;

}

else

{

if (Numbercontorls == 1) {

while (true) {

int numbersEmployee;

cout << "Введите количетсво сотрудников : ";

cin >> numbersEmployee;

Employee\* employees = new Employee[numbersEmployee];

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

cout << "Введите данные для сотрудника #" << (i + 1) << endl;

cout << "Введите имя: ";

cin.ignore();

getline(cin, employees[i].name);

cout << "Выберите должность (0 - Manager, 1 - Boss, 2 - Clerk, 3 - Salesman): ";

int vacancyChoice;

cin >> vacancyChoice;

employees[i].vacancy = static\_cast<Vacancies>(vacancyChoice);

cout << "Введите зарплату: ";

cin >> employees[i].salary;

cout << "Введите дату приема на работу (день, месяц, год): ";

cin >> employees[i].hireDate[0] >> employees[i].hireDate[1] >> employees[i].hireDate[2];

file << "Employees #" << i+1 << endl << employees[i];

}

cout << "Введите 0 для выхода, введите 1 для повторной записи сотрудников: ";

cin >> Numbercontorls;

if (Numbercontorls == 1)

{

continue;

}

else if (Numbercontorls == 0)

{

break;

}

delete[] employees;

}

}

else if (Numbercontorls == 2) {

cout << "Содержимое файла:" << endl;

string line;

while (getline(file, line)) {

cout << line << endl;

}

}

else if (Numbercontorls == 0) {

break;

}

else {

cout << "Введите правильное значение" << endl;

}

file.close();

}

}

return 0;

}

На рисунках 1-4 показаны скриншоты работающей программы:

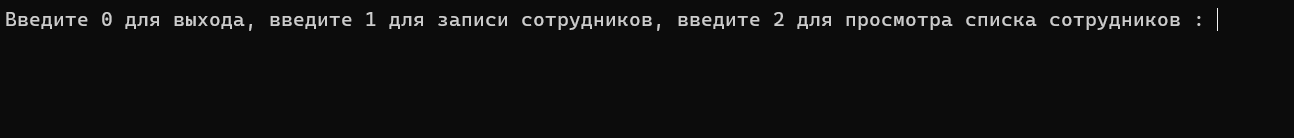


Рисунок 1 – окно выбора

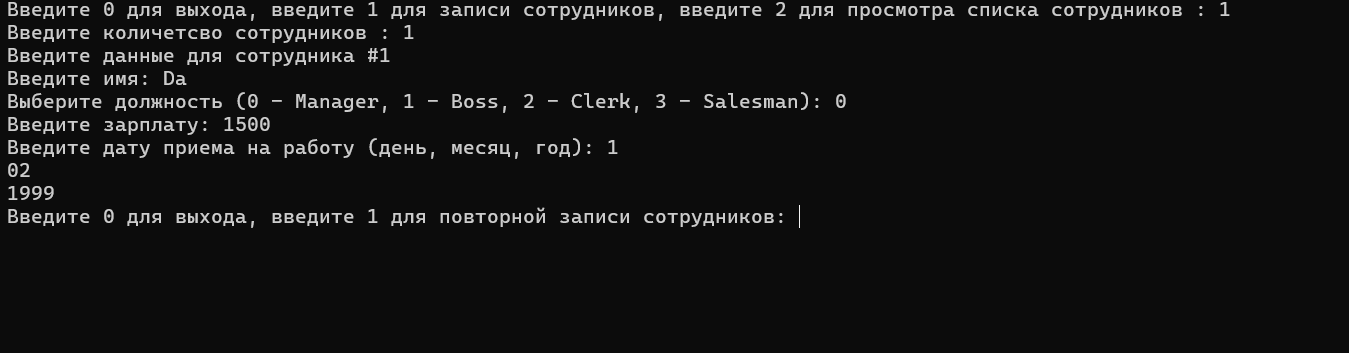


Рисунок 2 – окно ввода данных

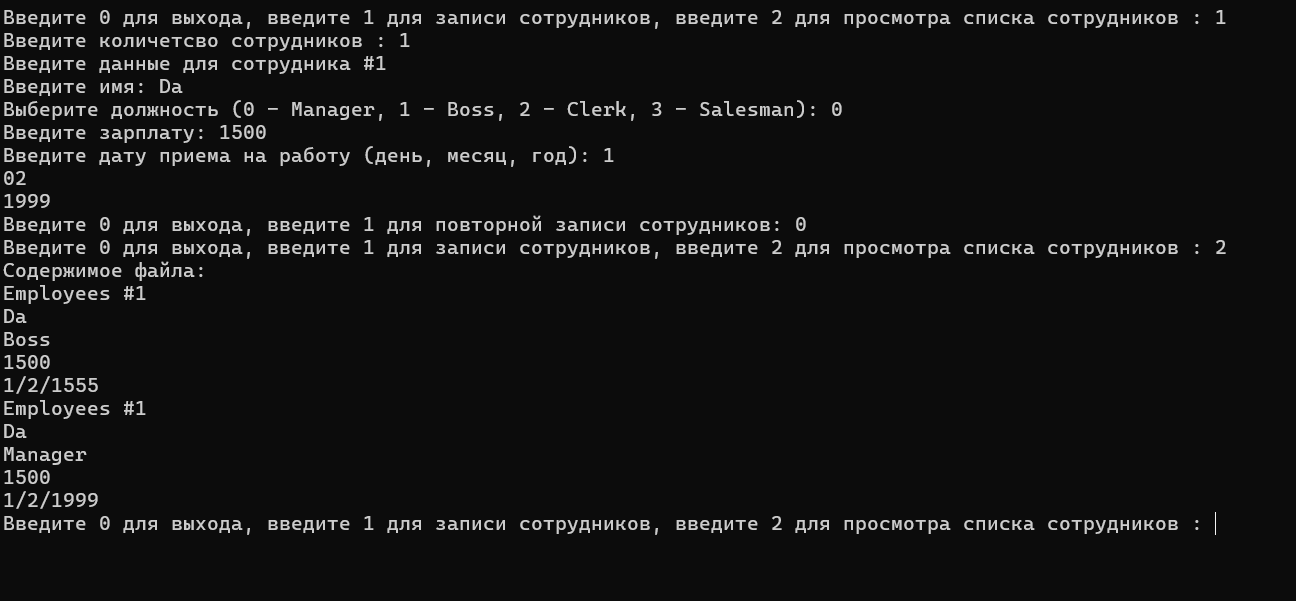
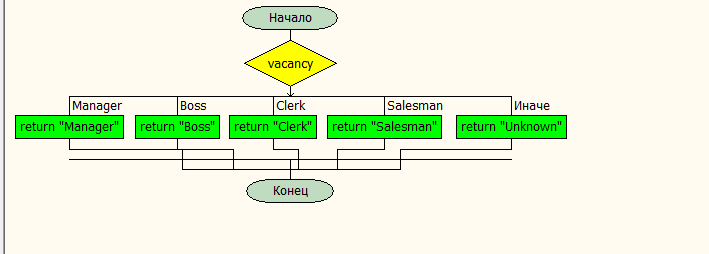
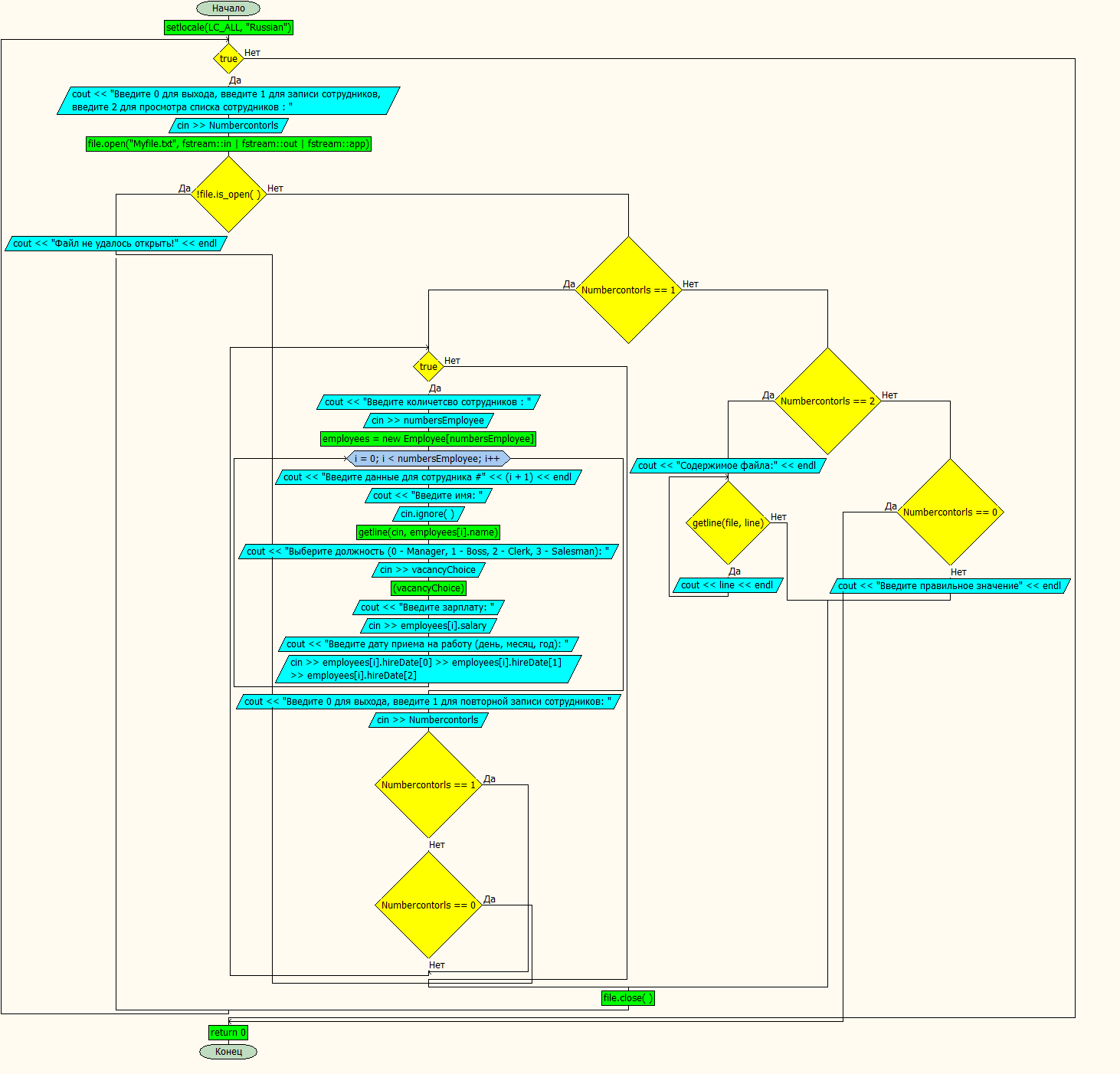


Рисунок 3 – окно просмотра данных

**Контрольные вопросы**

**Блок-схемы**

****

****

**Вывод**

В ходе выполнения лабораторной работы успешно создана программа. Которая способна создавать файл, записывать данные в файл и просматривать файл в консоли. Программа написана на языке С++.