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 No7

«Структуры и файлы»

Вариант 3

Подготовила: Бобровская А.Е.

Студент группы 410902

Проверил: Усенко Ф.В.

Минск 2024

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

Задание: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.

#include <iostream>

#include <fstream>

using namespace std;

enum Vacancies{

Manager,

Boss,

Clerk,

Salesman,

Other

};

struct Employee {

string name;

Vacancies vacancy;

int salary;

int date[3];

};

int main()

{

setlocale(LC\_ALL, "Russian");

int n, i, index;

cout << "Введите количество сотрудников: ";

cin >> n;

Employee\* Array = new Employee[n];

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

{

cout << "Введите имя " << i+1 << "-го сотрудника " << endl;

cin >> Array[i].name;

cout << "Введите должность сорудника (0 - manager, 1 - boss, 2 - clerk, 3 - saleman, 4 - other): ";

cin >> index;

Array[i].vacancy = static\_cast<Vacancies>(index);

cout << "Введите зарплату " << i+1 << "-го сотрудника" << endl;

cin >> Array[i].salary;

cout << "Введите дату трудоустройства " << i+1 << "-го сотрудника" << endl;

cin >> Array[i].date[0] >> Array[i].date[1] >> Array[i].date[2];

}

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

{

cout << "Имя сотрудника: " << Array[i].name << endl;

switch (Array[i].vacancy)

{

case Vacancies::Manager: cout << "Должность: " << "manager" << endl;

break;

case Vacancies::Boss: cout << "Должность: " << "boss" << endl;

break;

case Vacancies::Clerk: cout << "Должность: " << "clerk" << endl;

break;

case Vacancies::Salesman: cout << "Должность: " << "salesman" << endl;

break;

case Vacancies::Other: cout << "Должность: " << "other" << endl;

break;

}

cout << "Зарплата сотрудника: " << Array[i].salary << endl;

cout << "Дата трудоустройства: " << Array[i].date[0] << "." << Array[i].date[1] << "." << Array[i].date[2] << endl;

}

ofstream file;

file.open("Employee.txt");

if (!file.is\_open())

{

cout << "Ошибка открытия файла" << endl;

}

else

{

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

{

Employee person = Array[i];

file << "Имя: " << person.name << endl;

switch (Array[i].vacancy)

{

case Vacancies::Manager: file << "Должность: " << "manager" << endl;

break;

case Vacancies::Boss: file << "Должность: " << "boss" << endl;

break;

case Vacancies::Clerk: file << "Должность: " << "clerk" << endl;

break;

case Vacancies::Salesman: file << "Должность: " << "salesman" << endl;

break;

case Vacancies::Other: file << "Должность: " << "other" << endl;

break;

}

file << "Зарплата: " << person.salary << endl;

file << "Дата трудоустройства: " << person.date[0] << "." << person.date[1] << "." << person.date[2] << endl;

}

}

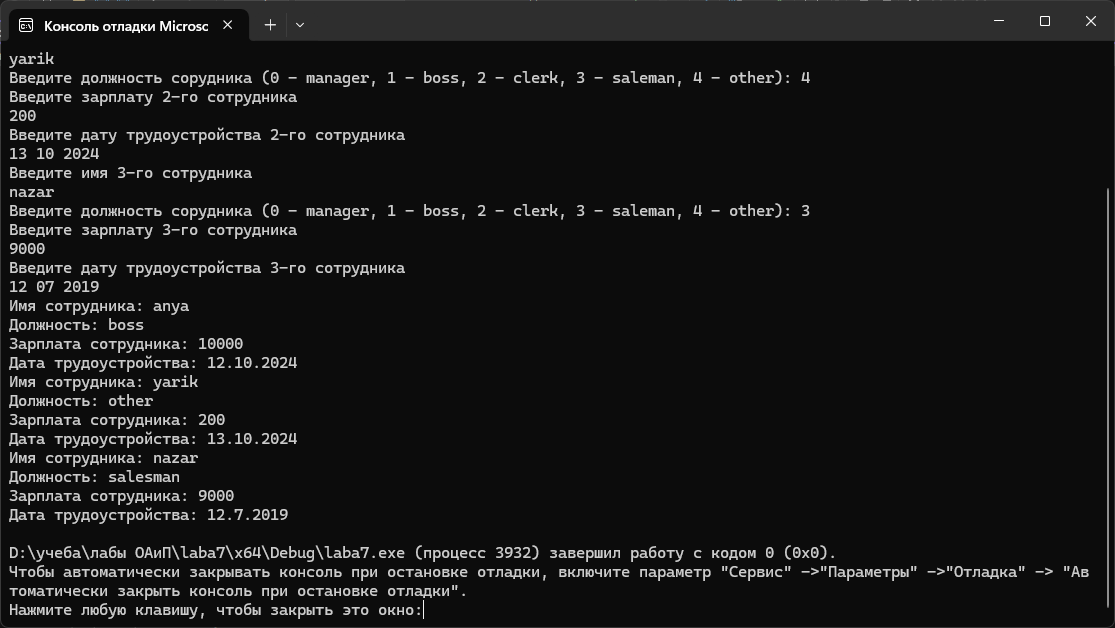
file.close();

delete Array;

return 0;

}

Результат работы программы представлен на рисунке 1, 2.



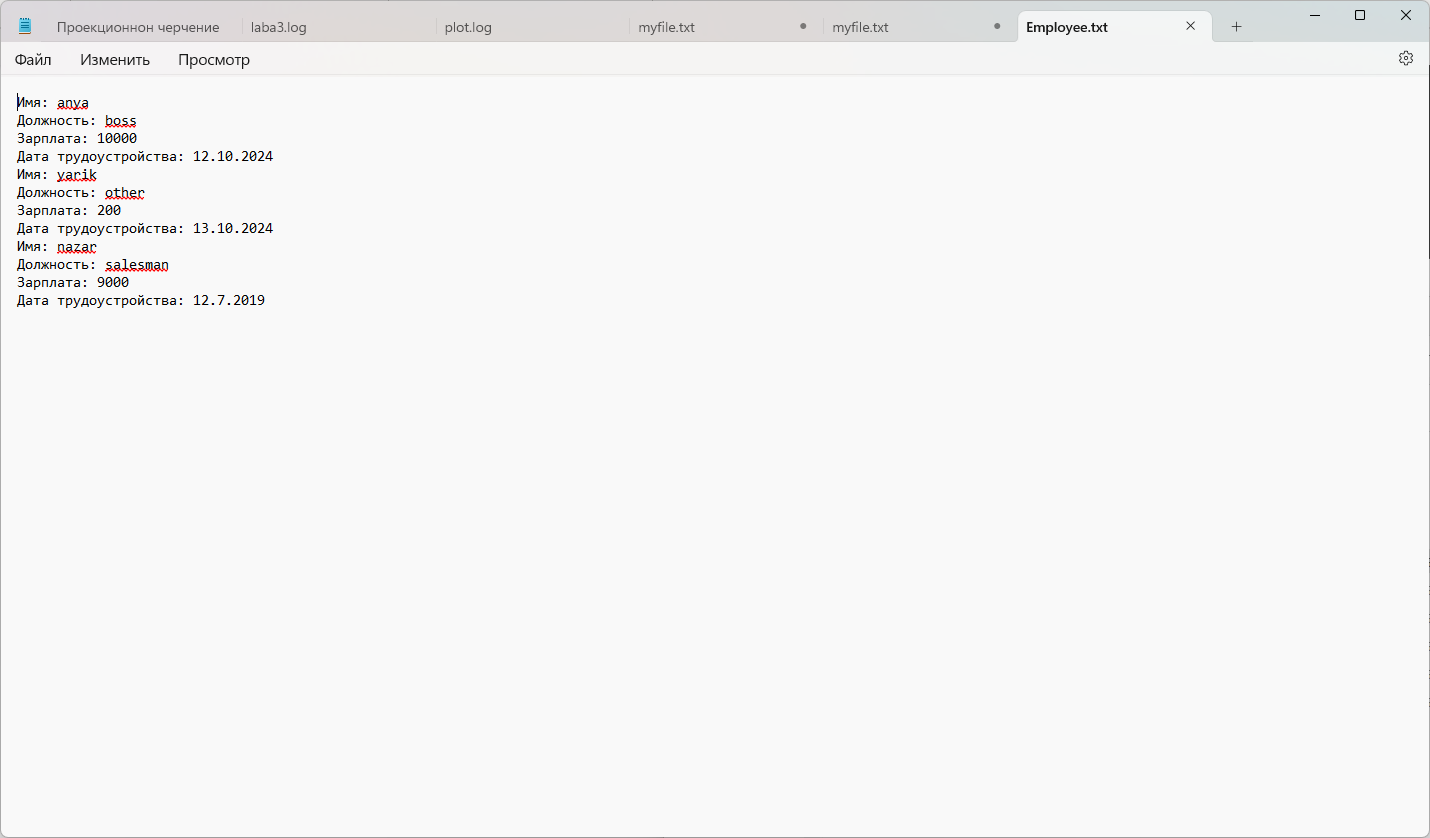


Рисунок 1, 2 – Результат работы программы

Блок-схема работы программы представлена на рисунке 2.

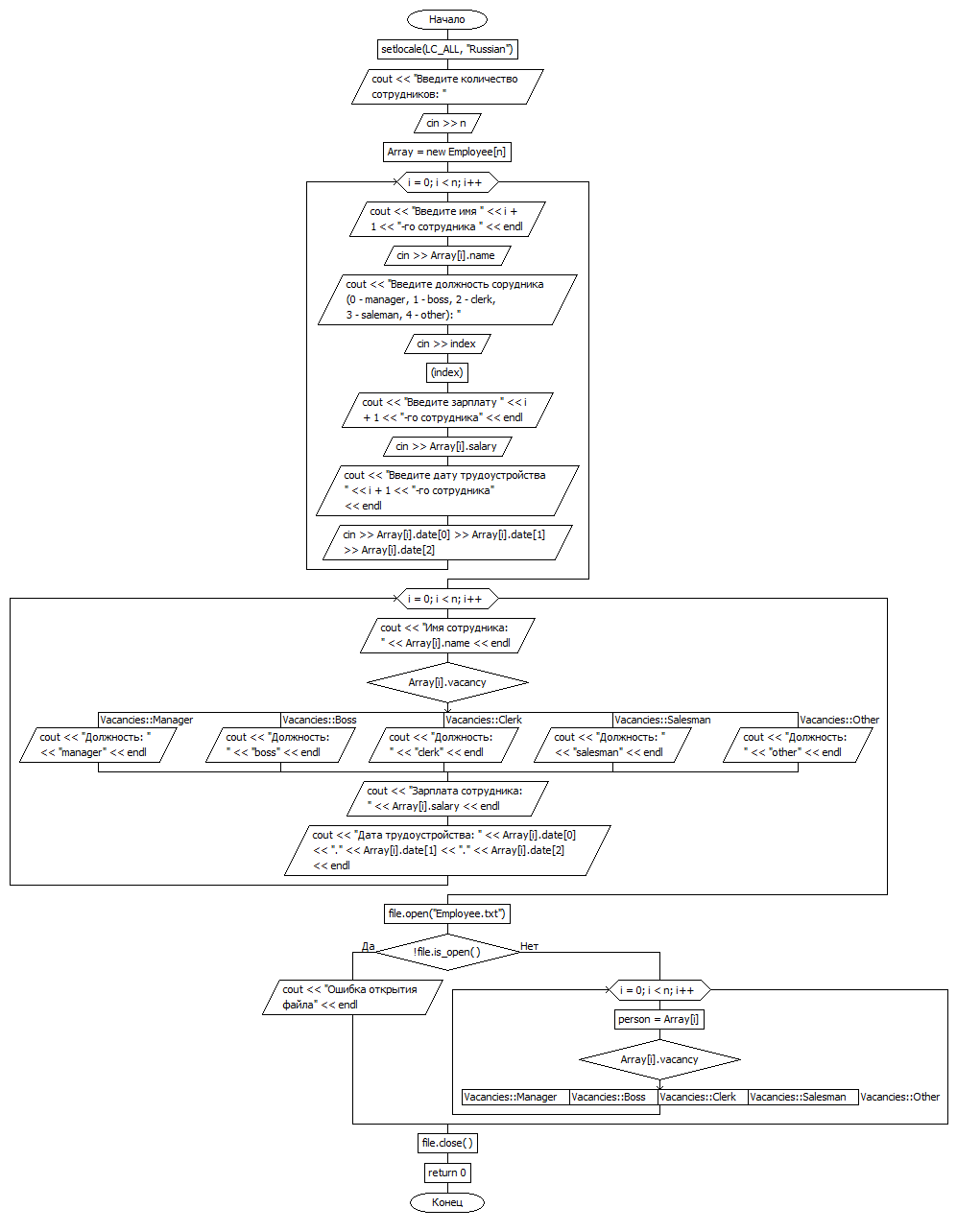


Рисунок 2 – блок схема работы программы

**Вывод:** в ходе выполнения работы была достигнута цель данной лабораторной работы: сформировать навыки и умения обработки структурированных типов данных, организованных в виде структур и файлов.