# ЛАБОРАТОРНАЯ РАБОТА №2

# «ПРОГРАММИРОВАНИЕ ОПЕРАЦИЙ НАД СТРУКТУРАМИ И БИНАРНЫМИ ФАЙЛАМИ»

Цель работы

Исследование особенностей обработки бинарных файлов, хранящих структурные типы данных.

Задания

1. Написать программу на языке C/C++, решающую задание варианта 12 из методических указаний к лабораторной работе;

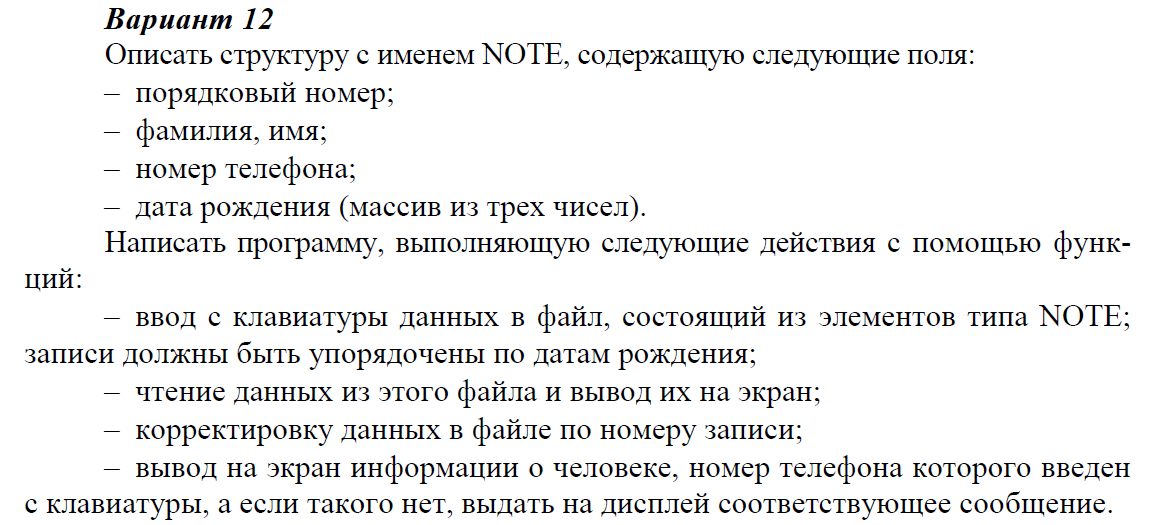


Рисунок 1 – Текст задания

1. Разработать структурные схемы алгоритма для всех функций решения задачи;
2. Разработать тестовые примеры и выполнить тестирование и отладку написанных программ;
3. Получить результаты работы программы и исследовать её свойства для различных режимов работы. Сформулировать выводы;

Текст программы

#include <stdio.h>

#include <string.h>

*//---------структуры и константы-------------*

*//структура для даты*

**struct** date {

**short** day; *//число*

**short** month; *//месяц*

**short** year; *//год*

};

*//структура для данных о человеке*

**struct** NOTE {

**int** n; *//номер*

**char** name[50]; *//имя*

**char** num[15]; *//номер телефона*

date birth; *//дата рождения*

};

*//константа величины структуры NOTE в байтах*

**const** **int** size\_note = **sizeof**(NOTE);

*//------------прототипы функций-------------*

**void** create\_file(FILE \*f);

**void** print\_file(FILE \*f);

**void** correct\_file(FILE \*f);

**void** search\_file(FILE \*f);

**void** sort\_file(FILE \*f);

*//-----главная функция - меню программы-----*

**int** main() {

*//указатель на файл*

FILE \*f=fopen("data.dat","r+b"); *//режим чтения*

**if** (!f) {

f=fopen("data.dat","w+b"); *//запись в пустой файл*

**if** (!f) {

puts("Can't open file'\n"); *//ошибка*

**return** 1;

}

}

**while** (1) {

*//текстовые указания к меню*

puts("----Menu----");

puts("1 - Input data");

puts("2 - Output data");

puts("3 - Correct data");

puts("4 - Person by phone number");

puts("5 - Sort data");

puts("0 - End");

**int** c;

printf("\nEnter command - ");

scanf("%d",&c);

*//вызов функций меню*

**switch** (c) {

**case** 1: create\_file(f); **break**; *//запись в файл*

**case** 2: print\_file(f); **break**; *//вывод содержимого файла*

**case** 3: correct\_file(f); **break**; *//корректировка содержимого*

**case** 4: search\_file(f); **break**; *//поиск по номеру телефона*

**case** 5: sort\_file(f); **break**; *//сортировка содержимого*

**case** 0: fclose(f); *//конец программы*

puts("\nThe End.");

**return** 0;

**default**: {

puts("\nNo such command\n"); *//команды не существует*

}

}

}

}

*//------------запись в файл-----------------*

**void** create\_file (FILE \*f) {

puts("\n----Enter data----");

NOTE name;

fseek(f,0,SEEK\_END);

**while** (1) {

printf("\n Number (0-exit) - "); *//номер*

scanf("%d",&name.n);

**if** (name.n==0) { *//конец записи*

printf("\n");

**return**;

}

printf(" Name - "); *//имя*

scanf("%s",&name.name);

printf(" Phone number - "); *//телефон*

scanf("%s",&name.num);

printf(" Day - "); *//день*

scanf("%d",&name.birth.day);

printf(" Month - "); *//месяц*

scanf("%d",&name.birth.month);

printf(" Year - "); *//год*

scanf("%d",&name.birth.year);

fwrite(&name,size\_note,1,f);

}

}

*//--------------вывод файла-----------------*

**void** print\_file(FILE \*f) {

puts("\n----Print data----");

NOTE name;

**int** n;

rewind(f); *//указатель в начало файла*

**do** {

n=fread(&name,size\_note,1,f); *//проверка конца файла*

**if** (n<1) **break**;

printf("\n");

printf(" Number - %d\n",name.n); *//номер*

printf(" Name - %s\n",name.name); *//имя*

printf(" Phone number - %s\n",name.num); *//телефон*

printf(" Birthday - "); *//дата рождения*

*//если день или месяц <10, добавить спереди 0*

**if** (name.birth.day>9)

printf("%d.",name.birth.day);

**else** printf("0%d.",name.birth.day);

**if** (name.birth.month>9)

printf("%d.",name.birth.month);

**else** printf("0%d.",name.birth.month);

printf("%d\n",name.birth.year);

} **while** (1);

printf("\n");

**return**;

}

*//------корректировка по номеру записи-------*

**void** correct\_file(FILE \*f) {

printf("\n----Correct by number----\n");

**int** n, s;

NOTE name;

printf("\nEnter number - "); *//ввод телефона для поиска*

scanf("%d",&s);

rewind(f);

**while** (1) {

n=fread(&name,size\_note,1,f); *//проверка конца файла*

**if** (n<1) **break**;

**if** (s==name.n) { *//изменение информации*

fseek(f,(-1)\*size\_note,SEEK\_CUR);

printf("\n Number - "); *//номер*

scanf("%d",&name.n);

printf(" Name - "); *//имя*

scanf("%s",&name.name);

printf(" Phone number - "); //телефон

scanf("%s",&name.num);

printf(" Day - "); *//день*

scanf("%d",&name.birth.day);

printf(" Month - "); *//месяц*

scanf("%d",&name.birth.month);

printf(" Year - "); *//год*

scanf("%d",&name.birth.year);

fwrite(&name,size\_note,1,f);

puts("\nDone successfully\n");

**return**;

}

}

printf("\nNo such number\n\n"); *//ошибка номера*

**return**;

}

*//------------поиск в файле------------------*

**void** search\_file(FILE \*f) {

printf("\n----Search in file----\n");

NOTE name; *//прочитанная запись*

NOTE searched; *//искомая запись*

searched.n=0;

**char** phone[12];

**int** n;

printf("\nEnter number - "); *//ввод номера телефона*

scanf("%s",&phone);

rewind(f);

**while** (1) {

n=fread(&name,size\_note,1,f); *//проверка конца файла*

**if** (n<1) **break**;

**if** (strcmp(phone,name.num)==0) { *//сравнение номеров*

searched=name;

**break**;

}

}

*//вывод информации о человеке с искомым номером*

**if** (searched.n!=0) {

printf("\nInformation:\n");

printf(" Number - %d\n",searched.n);

printf(" Name - %s\n",searched.name);

printf(" Phone number - %s\n",searched.num);

printf(" Birthday - ");

**if** (searched.birth.day>9)

printf("%d.",searched.birth.day);

**else** printf("0%d.",searched.birth.day);

**if** (searched.birth.month>9)

printf("%d.",searched.birth.month);

**else** printf("0%d.",searched.birth.month);

printf("%d\n\n",searched.birth.year);

}

**else**

*//сообщение об ошибке*

printf("\nNo such person\n\n");

**return**;

}

*//------------сравнение дат------------------*

**int** datecmp(date d1, date d2) {

**if** (d1.year>d2.year) *//сравнение годов*

**return** 1;

**if** (d1.month>d2.month) *//сравнение месяцев*

return 1;

**if** (d1.day>d2.day) *//сравнение дней*

**return** 1;

**return** 0;

}

*//----------сортировка по дате---------------*

**void** sort\_file(FILE \*f) {

printf("\n----Sort data----");

fseek(f,0,SEEK\_END);

**int** l = ftell(f)/size\_note; *//количество записей*

rewind(f);

**int** i, j;

NOTE n1, n2, n3;

*//сортировка прямым обменом*

**for** (i=1;i<l;i++)

**for** (j=0;j<l-i;j++) {

fseek(f,j\*size\_note,SEEK\_SET);

*//чтение двух записей*

fread(&n1,size\_note,1,f);

fread(&n2,size\_note,1,f);

**if** (datecmp(n1.birth,n2.birth)==1) {

*//перестановка местами*

fseek(f,(-2)\*size\_note,SEEK\_CUR);

fwrite(&n2,size\_note,1,f);

fwrite(&n1,size\_note,1,f);

}

}

puts("\nSorted successfully\n");

**return**;

}

Структурная схема программы

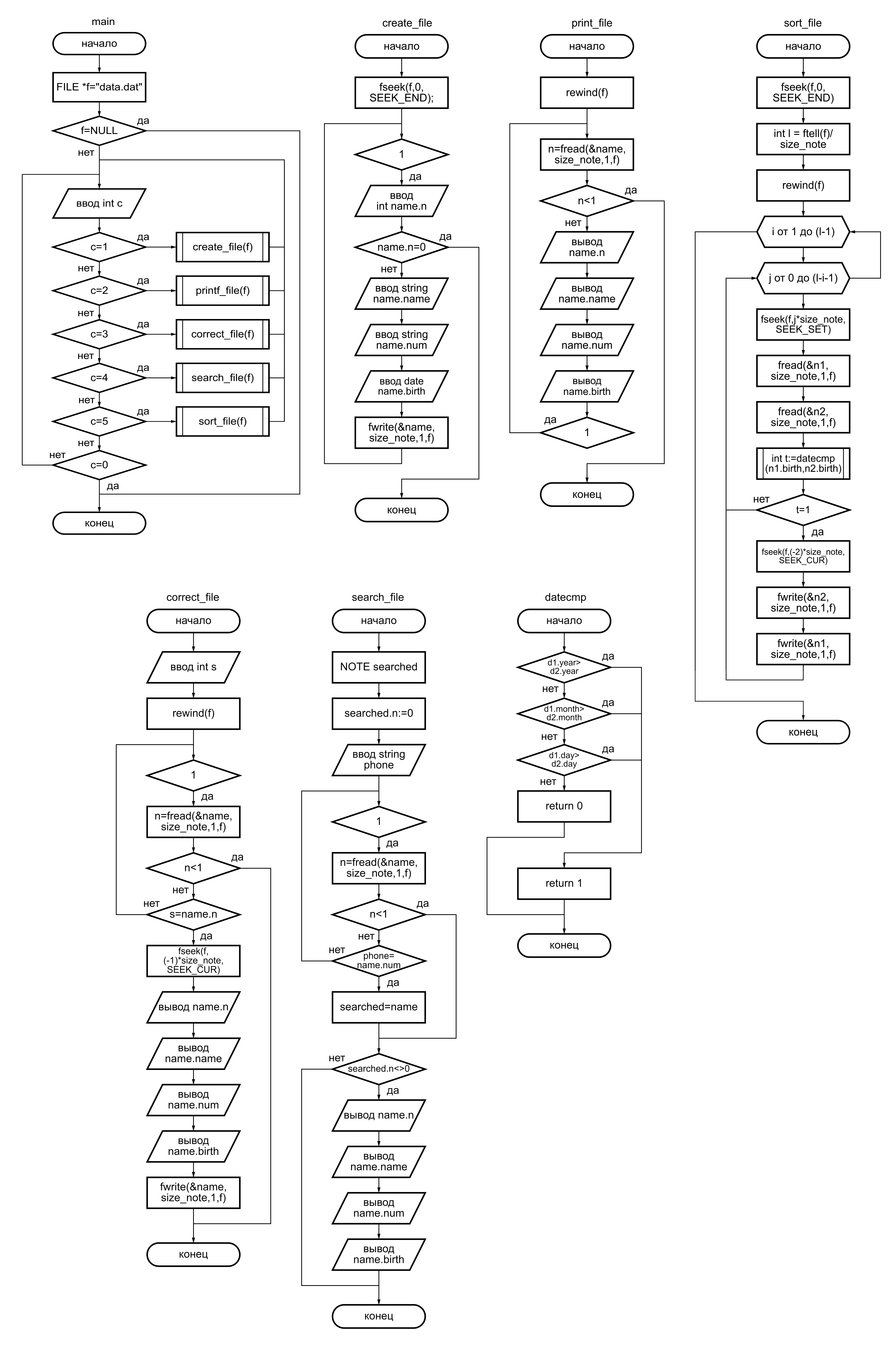


Рисунок 2 – Структурная схема программы

Тестовые примеры

С помощью программы в пустой файл data.dat, хранящийся в той же папке, что и программа, введено три структуры типа NOTE.

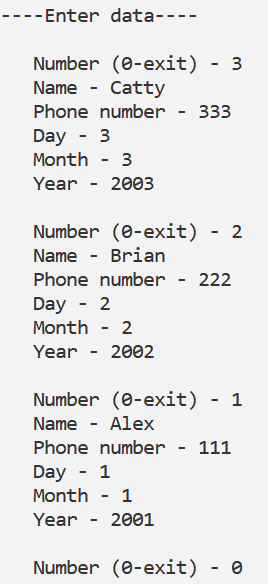


Рисунок 3 – Ввод информации

Для проверки правильности ввода и вывода вызвана функция чтения и вывода информации из файла. Видно, что данные были записаны в файл верно.

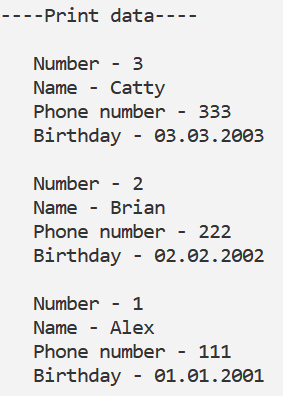


Рисунок 4 – Вывод информации

Функция заполнения файла вызвана повторно для ввода новой записи.

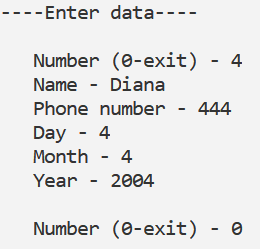


Рисунок 5 – Новая запись

Для проверки повторного ввода вызвана функция вывода данных из файла. Видно, что в результате повторного ввода новая запись была добавлена в конец файла с ранее записанными данными.

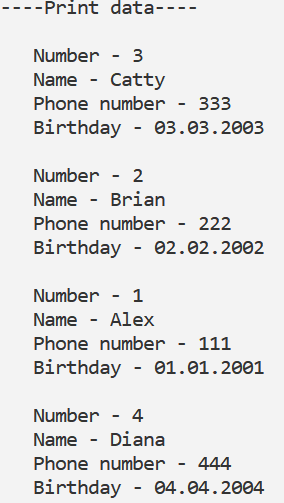


Рисунок 6 – Вывод обновлённой информации

Была вызвана функция сортировки и функция вывода данных из файла. В результате видно, что записи в файле были отсортированы по убыванию даты рождения.

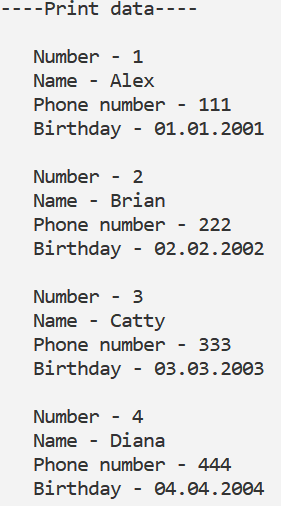


Рисунок 7 – Вывод отсортированных данных

Проверка функции коррекции данных. В записи с номером 1 проведена замена имени.

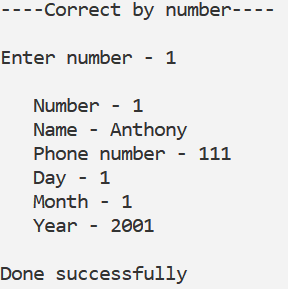


Рисунок 8 – Изменение записи

В результате вывода данных видно, что изменение проведено успешно.

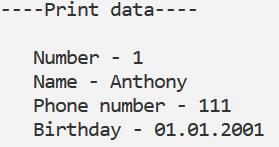


Рисунок 9 – Вывод изменённой записи

Проверка функции поиска внутри файла. В результате поиска по номеру телефона 333 программа вывела верную запись.

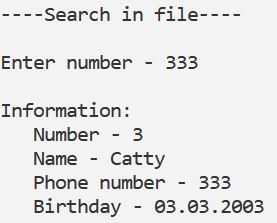


Рисунок 10 – Вывод искомой записи

Функция поиска вызвана повторно. При вводе отсутствующего в записях файла номера 0 программа выдала сообщение об отсутствии номера.

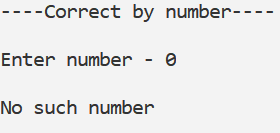


Рисунок 11 – Сообщение об ошибке

При вызове команды завершения программы программа была остановлена.

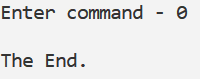


Рисунок 12 – Завершение программы

По результатам проверки программы можно заключить, что она написана верно и успешно справляется с поставленными перед ней задачами.

Вывод

В ходе работы были изучены основы работы со структурами и бинарными файлами в языках С/С++. Освоены методы обработки бинарных файлов, содержащих данные структурного типа. Результатом работа стала программа, выполняющая заполнение, чтение, вывод, коррекцию, поиск и сортировку данных внутри бинарного файла.