Министерство образования Республики Беларусь

Учреждение образования

«Брестский Государственный технический университет»

Кафедра ИИТ

**Лабораторная работа №11**

По дисциплине «Основы алгоритмизации и программирования»

Тема: «Бинарные и текстовые файлы»

**Выполнил:**

Студент 1 курса

Группы ИИ-21

Карагодин Д.Л.

**Проверил:**

Гирель Т.Н

Брест 2022

**Цель работы:** Изучить принципы программирования с использованием бинарных и текстовых файлов. Ознакомиться с основными функциями в Си для работы с бинарными и текстовыми файлами.

**Вариант №1**

**Текст задания**

**В программу разработанную в лабораторной работе 10 добавить чтение и сохранение данных массива структур при помощи бинарных и текстовых файлов следующим образом:**

1. **При первом запуске программы должен создаваться бинарный или текстовый файл на выбор пользователя для хранения данных из массива структур.**
2. **При добавлении новой записи в массив структур в файл должна дописываться новая запись, без изменения остальных записей.**
3. **При повторном запуске программы, если файл уже существует, то информация в массив структур должна читаться из этого файла. Если файл отсутствует, то он должен создаваться (см. Пункт 1).**
4. **Все изменения (сортировка, изменения полей записи, удаление записи) – сохраняются в файле при помощи полной перезаписи содержимого.**
5. **Сделать вывод о том, какие преимущества использования конкретного типа файлов (бинарные или текстовые) в решаемой вами задаче.**

**Текст программы**

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

#include <string.h>

#include <malloc.h>

#pragma pack(push, 1)

struct data {

char name[15];

char surname[15];

char middle\_name[15];

char local\_street[20];

short discount;

};

#pragma pop

short razmer = 0;

struct data\* people;

FILE\* file;

char typefile[3];

char name[30];

void output\_struct() {

printf("%-5s %-15s %-15s %-15s %-20s %-15s\n", "Номер", "Имя", "Фамилия", "Отчество", "Улица проживания", "Размер скидки");

for (short i = 0; i < razmer; i++) {

printf(" %-3d", i + 1);

printf(" %-15s", people[i].name);

printf(" %-15s", people[i].surname);

printf(" %-15s", people[i].middle\_name);

printf(" %-20s", people[i].local\_street);

printf(" %-d%%\n", people[i].discount);}

printf("\n\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n");}

void input\_struct() {

++razmer;

people = (struct data\*)realloc(people, razmer \* sizeof(struct data));

printf("Введите данные:\n");

printf("Имя: ");

scanf("%s", people[razmer - 1].name);

printf("Фамилия: ");

scanf("%s", people[razmer - 1].surname);

printf("Отчество: ");

scanf("%s", people[razmer - 1].middle\_name);

printf("Улица проживания: ");

scanf("%\*c");

scanf("%[^\n]", people[razmer - 1].local\_street);

printf("Размер скидки в процентах: ");

scanf("%hd", &people[razmer - 1].discount);

printf("\n\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n");}

void procent\_5() {

for (short i = 0; i < razmer; i++) {

printf("%-5s %-15s %-15s %-15s %-20s %-15s\n", "Номер", "Имя", "Фамилия", "Отчество", "Улица проживания", "Размер скидки");

if (people[i].discount == 5) {

printf(" %-3d", i + 1);

printf(" %-15s", people[i].name);

printf(" %-15s", people[i].surname);

printf(" %-15s", people[i].middle\_name);

printf(" %-20s", people[i].local\_street);

printf(" %-d%%\n", people[i].discount);}}

printf("\n\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n");}

void delete\_element() {

short id;

output\_struct();

printf("Введите id элемента который нужно удалить: ");

scanf("%hd", &id);

struct data buffer;

for (short i = 0; i < razmer - id; i++) {

buffer = people[i+id];

people[i+id] = people[i+id - 1];

people[i+id - 1] = buffer;}

razmer--;

people = (struct data\*)realloc(people, razmer \* sizeof(struct data) );

printf("\n\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n");}

void sort() {

short choise, maxmin, id;

struct data buffer;

printf("Хотите сортировать\nПо возрастанию -> 1\nИли убыванию -> 2 ?\nВаш выбор: ");

scanf("%hd", &choise);

if (choise == 1) {

for (short i = 0; i < razmer; i++) {

maxmin = 100; id = 0;

for (short j = i; j < razmer; j++) {

if (people[j].discount <= maxmin) { maxmin = people[j].discount; id = j; }

}

buffer = people[id];

people[id] = people[i];

people[i] = buffer;}}

else {

for (short i = 0; i < razmer; i++) {

maxmin = 0; id = 0;

for (short j = i; j < razmer; j++) {

if (people[j].discount >= maxmin) { maxmin = people[j].discount; id = j; }

}

buffer = people[id];

people[id] = people[i];

people[i] = buffer;}}

printf("\n\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n");}

void fread() {

if (strcmp(typefile, "bin") != 0) {

printf("fread (txt) <- запустилась \n");

while (!feof(file)) {

++razmer;

people = (struct data\*)realloc(people, razmer \* sizeof(struct data));

fscanf(file,"%15s", people[razmer - 1].name);

fscanf(file,"%15s", people[razmer - 1].surname);

fscanf(file,"%15s", people[razmer - 1].middle\_name);

fscanf(file, "%\*c");

fscanf(file,"%[^\n]", people[razmer - 1].local\_street);

fscanf(file,"%hd", &people[razmer - 1].discount);

printf("razmer = %hd. \n",razmer);}}

else {

printf("fread (bin) <- запустилась\n");

int a,count=0; fread(&a,sizeof(int),1,file);

printf("a = %d\n", a);

printf("data = %d\n", sizeof(struct data));

while (count<a) {

++razmer;

printf("razmer = %hd.\n", razmer);

people = (struct data\*)realloc(people, razmer \* sizeof(struct data));

fread(people[razmer - 1].name,sizeof(char), 15,file);

printf("%d ", ftell(file));

fread(people[razmer - 1].surname, sizeof(char), 15,file);

printf("%d ", ftell(file));

fread(people[razmer - 1].middle\_name, sizeof(char), 15,file);

printf("%d ", ftell(file));

fread(people[razmer - 1].local\_street, sizeof(char), 20,file);

printf("%d ", ftell(file));

fread(&people[razmer - 1].discount,sizeof(short),1,file);

printf("%d \n", ftell(file));

count += sizeof(struct data);}}}

void fwrite() {

rewind(file);

short count = 0;

if (strcmp(typefile, "bin") != 0) {

printf("fwrite (txt) <- запустилась\n");

file = fopen(name, "w+t");

while (count<razmer) {

fprintf(file, "%s\n", people[count].name);

fprintf(file, "%s\n", people[count].surname);

fprintf(file, "%s\n", people[count].middle\_name);

fprintf(file, "%s\n", people[count].local\_street);

if(count!=razmer-1) fprintf(file, "%hd\n",people[count].discount);

else fprintf(file, "%hd", people[count].discount);

count++;}}

else {

printf("fwrite (bin) <- запустилась\n");

file = fopen(name, "wb+");

int a = sizeof(struct data) \* razmer;

fwrite(&a, sizeof(int), 1, file);

while (count < razmer) {

fwrite(people[count].name, sizeof(char), 15, file);

printf("%d ", ftell(file));

fwrite(people[count].surname, sizeof(char), 15, file);

printf("%d ", ftell(file));

fwrite(people[count].middle\_name, sizeof(char), 15, file);

printf("%d ", ftell(file));

fwrite(people[count].local\_street, sizeof(char), 20, file);

printf("%d ", ftell(file));

fwrite(&people[count].discount, sizeof(short), 1, file);

printf("%d \n", ftell(file));

count++;}}}

void init() {

people = (struct data\*)calloc(sizeof(struct data), razmer);

printf("Введите имя файла(с txt или bin): ");

scanf("%s", name);

char name\_buffer[30];

strcpy(name\_buffer, name);

char \*typefl=strtok(name\_buffer,".");

typefl = strtok(NULL, ".");

typefile[0] = typefl[0];

typefile[1] = typefl[1];

typefile[2] = typefl[2];

if ((file = fopen(name, "r")) == NULL) {

printf("Так как файла не было,мы его создали\n");

if (strcmp(typefile, "bin") != 0) {file = fopen(name, "wt+"); }

else { file = fopen(name, "wb+"); }}

else{

if (strcmp(typefile, "bin") != 0) { file = fopen(name, "rt+"); fread();}

else { file = fopen(name, "rb+"); fread(); }}}

int main() {

init();

short Exit = 0;

short choise = 0;

while (Exit == 0) {

printf("Что вы собираетесь сделать?\n1.Добавить новый элемент структуры\n2.Вывести всю таблицу\n3.Вывести всех с скидкой 5%\n4.Сортировка по скидке\n5.Удалить элемент структуры\n6.Выйти из программы\nВаш выбор: ");

scanf("%hd", &choise);

printf("\n\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n");

if (choise == 1) input\_struct();

if (choise == 2) output\_struct();

if (choise == 3) procent\_5();

if (choise == 4) sort();

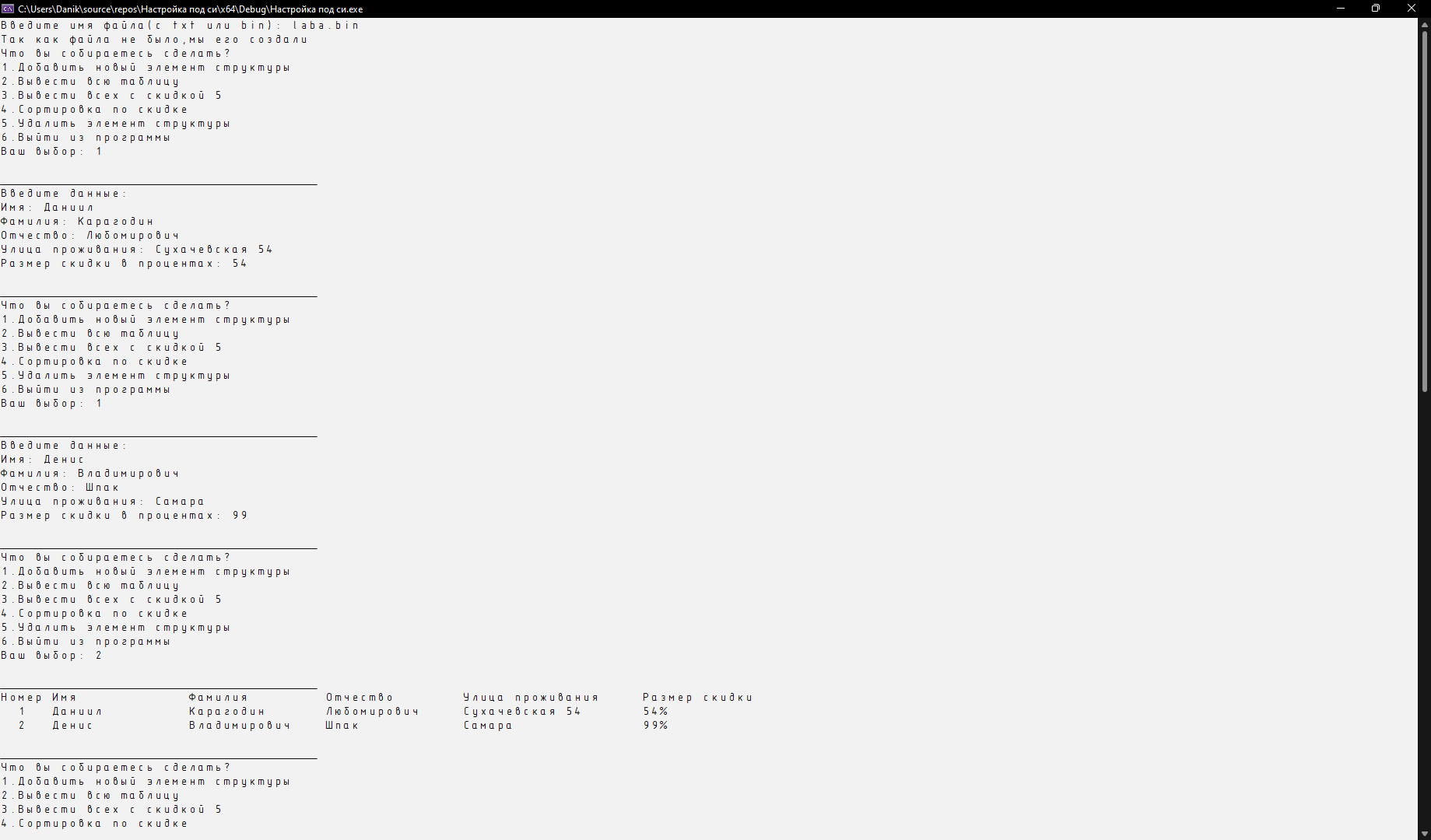
if (choise == 5) delete\_element();

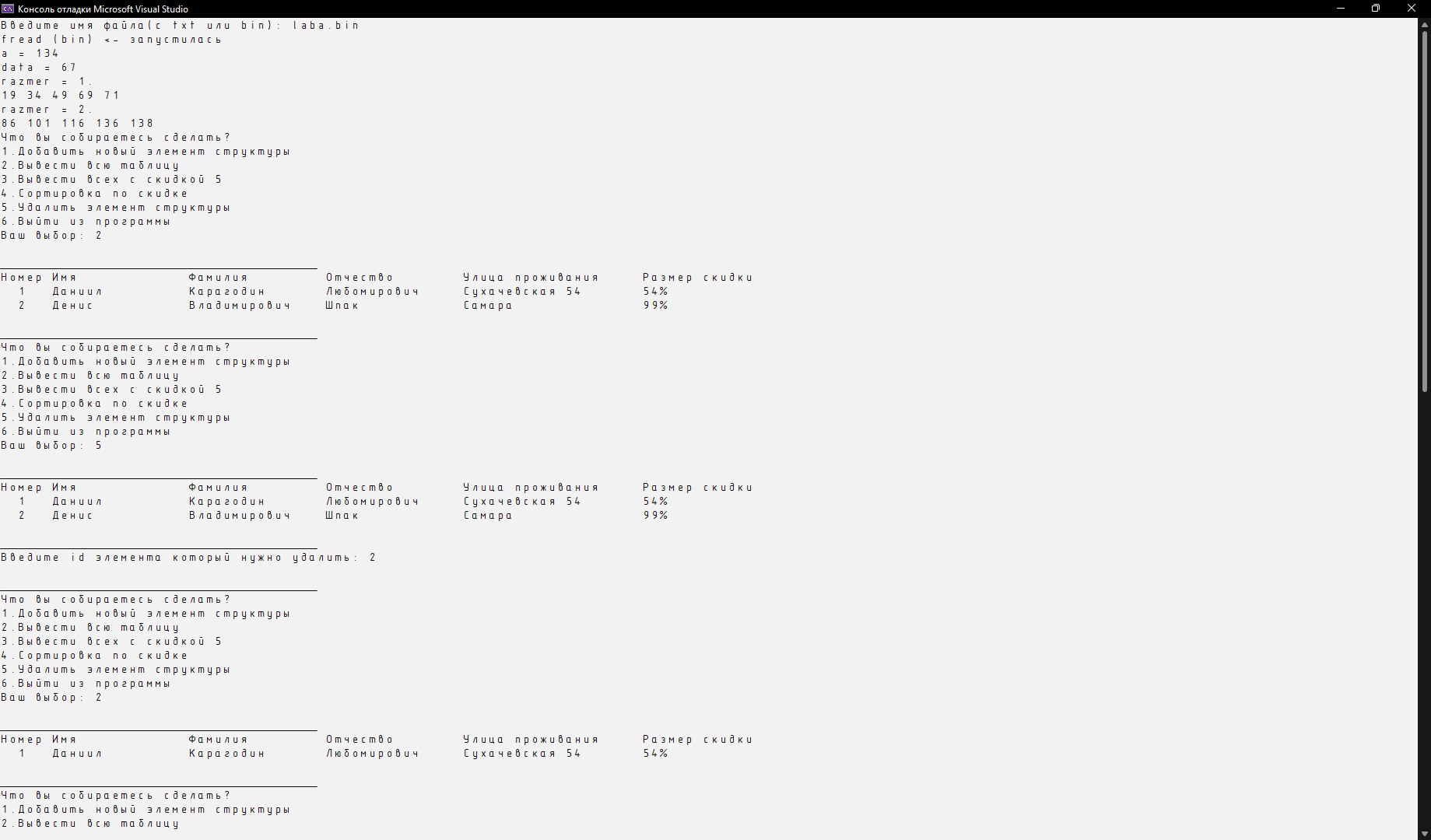
if (choise == 6) Exit = 1;}

fwrite();

\_fcloseall();

return 0;}

**Результат работы**



У бинарного файла есть следующие преимущества: конфиденциальность файлов(нельзя прочитать информацию без специальных программ),более быстрый процесс чтения файла.

У текстового файла есть следующие преимущества: малый размер файлов,переносимость файлов и их информаций на другие системы.

**Вывод:** Изучил принципы программирования с использованием бинарных и текстовых файлов. Ознакомился с основными функциями в Си для работы с бинарными и текстовыми файлами.