Постановка завдання

Сформувати двійковий файл із елементів, заданої у варіанті структури, роздрукувати його вмістиме, виконати знищення й додавання елементів у відповідності зі своїм варіантом, використовуючи для пошуку елементів що знищуються чи додаються, функцію. Формування, друк, додавання й знищення елементів оформити у вигляді функцій. Передбачити повідомлення про помилки при відкритті файлу й виконанні операцій вводу/виводу.

//8. Структура "Покупець":

//-прізвище, ім'я, по батькові;

//- домашня адреса;

//-номер телефону;

//-номер кредитної картки.

//Знищити 3 елементи з початку файлу, додати 3 елементи в кінець файлу.

//3. Текст функцій для :

//• формування файлу,

//• друку файлу,

//• додавання запису у файл,

//• знищення запису з файлу

//• пошуку структури для знищення.

#include"pch.h"

#include <stdio.h>

#include <string.h>

#include <stdlib.h>

#pragma warning(disable:4996)

//умовна к-сть структур

const int AMOUNT\_OF\_STRUCT = 4;

typedef struct customer

{

char surname[20];

char name[20];

char pobatkovi[20];

char homeAdress[20];

char phoneNumber[20];

char creditNumber[20];

} customer\_t;

//зафілюю структури

customer\_t FillStruct()

{

customer buyer;

printf("Surname:\t"); scanf("%s", &buyer.surname);

printf("Name:\t"); scanf("%s", &buyer.name);

printf("Pobatkovi:\t"); scanf("%s", &buyer.pobatkovi);

printf("Home Adress:\t"); scanf("%s", &buyer.homeAdress);

printf("Phone Number:\t"); scanf("%s", &buyer.phoneNumber);

printf("Number of credit card:\t"); scanf("%s", &buyer.creditNumber);

return buyer;

}

void WriteStructToFile(customer \* structArray, const char\* file, int is\_append)

{

FILE \*f;

if (!is\_append)

{

f = fopen(file, "w+b");

}

else

{

f = fopen(file, "a");

}

if (f == NULL)

{

printf("File wasn't opened correctly");

exit(1);

}

for (int i = 0; i < AMOUNT\_OF\_STRUCT; i++)

{

fwrite(&structArray[i], sizeof(customer), 1, f);

}

fclose(f);

}

void fileRead(customer \*structArray1, const char\*file)

{

FILE \*f; int i = 0; f = fopen(file, "r");

while (!feof(f) && i < AMOUNT\_OF\_STRUCT)

{

fread(&structArray1[i], sizeof(customer), 1, f);

i++;

}

}

int structureToDelete = 3;

//видалення структури

void DeleteElements(customer \*structureArray2, customer \*structureArray1)

{

int structureToLeave = AMOUNT\_OF\_STRUCT - structureToDelete;

for (int i = 0; i < structureToLeave; i++)

{

structureArray2[i] = structureArray1[structureToDelete + i];

}

}

//додавання структури

void AddElements(customer \*structureArray2)

{

for (int i = AMOUNT\_OF\_STRUCT - structureToDelete; i < AMOUNT\_OF\_STRUCT; i++)

{

structureArray2[i] = FillStruct();

}

}

void main()

{

customer structureArray[AMOUNT\_OF\_STRUCT];

customer structureArray1[AMOUNT\_OF\_STRUCT];

customer structureArray2[AMOUNT\_OF\_STRUCT];

for (int i = 0; i < AMOUNT\_OF\_STRUCT; i++)

{

printf("CUSTOMER:\n");

structureArray[i] = FillStruct();

}

WriteStructToFile(structureArray, "f.txt", 0);

fileRead(structureArray1, "f.txt");

DeleteElements(structureArray2, structureArray1);

AddElements(structureArray2);

WriteStructToFile(structureArray2, "f.txt", 0);

//вивід зміненого порядку структур

for (int i = 0; i < AMOUNT\_OF\_STRUCT; i++)

{

printf("surname - %s\n", structureArray2[i].surname);

printf("name - %s\n", structureArray2[i].name);

printf("pobatkovi - %s\n", structureArray2[i].pobatkovi);

printf("Home adress - %s\n", structureArray2[i].homeAdress);

printf("Phone number - %s\n", structureArray2[i].phoneNumber);

printf("Number of credit card - %s\n", structureArray2[i].creditNumber);

printf("\n");

}

}



