Фаховий коледж ракетно-космічного машинобудування

Дніпровського національного університету імені Олеся Гончара

ЗВІТ

з лабораторних робіт

з дисципліни «Технології (Програмування)»

Спеціальність 123 Комп’ютерна інженерія

Група КС-21-1

Виконав А.С. Василенко

Перевірив М.М. Гапоненко

2022-2023

ЗМІСТ

Лабораторна робота № 16

ЛАБОРАТОРНА РОБОТА № 16

Мета: Придбати практичні навички при складання програм роботи з файлами структур з використанням функцій fread(), fwrite(), fseek() та ftell().

Хід роботи

1 Постановка задачі.

Загальна постановка завдання:

Створити та реалізувати обробку файлу структур даних згідно Вашого варіанта. Програма повинна задовольняти наступним вимогам:

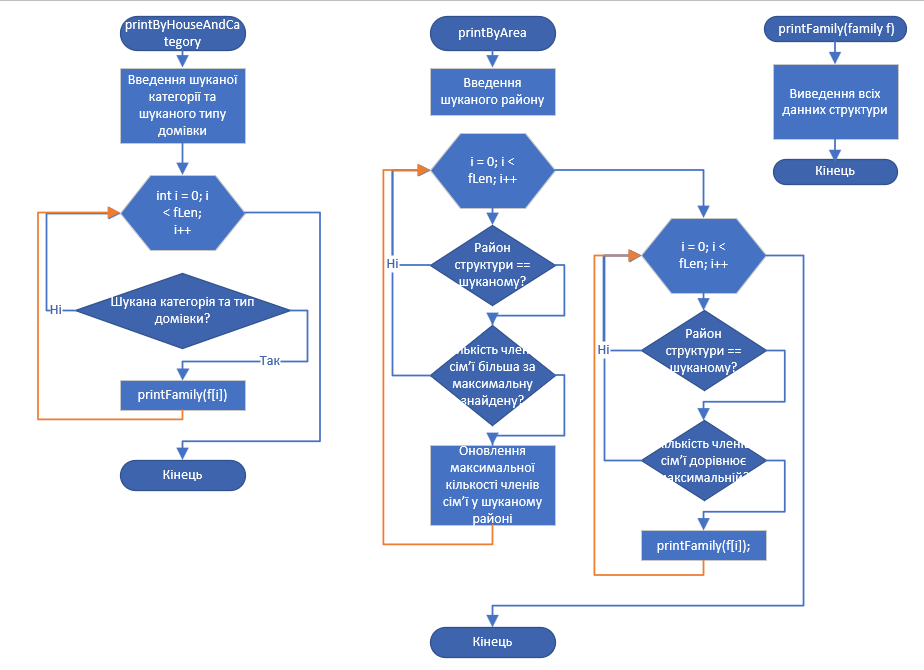
− повинні бути реалізовані наступні функції обробки файлу структур:

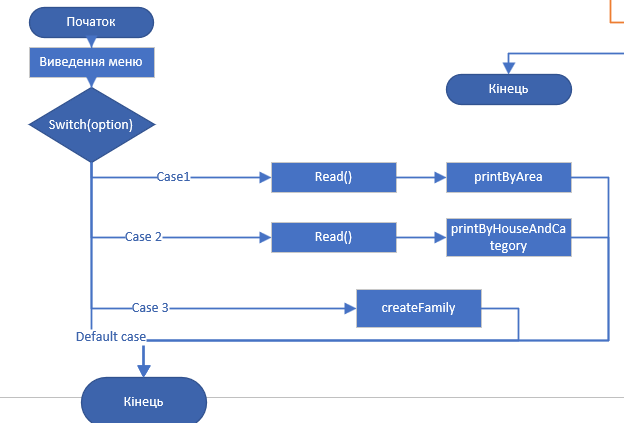
1. перегляд вмісту файлу;
2. реалізація запитів до файлу згідно з індивідуальними завданнями Вашого варіанту.

* організувати меню для вибору користувачем варіанту виконання вказаних вище функцій. Забезпечити повернення в меню програми після виконання чергового пункту меню. Вихід із програми здійснювати по відповідному пункту меню: «Вихід з програми»;
* при обробці файлу враховувати, що шукані елементи можуть бути відсутні. В цьому випадку вивести користувачеві відповідне повідомлення;

− введення та виведення вхідних та вихідних даних повинно містити необхідні для користувача повідомлення.

|  |  |  |
| --- | --- | --- |
| 3 | Скласти програму, що містить відомості про соціально незахищені сім’ї міста.  Структура повинна складатися з полів:   * Категорія – малозабезпечений, чорнобилець, інвалід(група), багатодітний; * Кількість членів у сім’ї; * Район мешкання; * Вид помешкання – особистий дім, багатоповерхівка, барак, без помешкання. | Видати наступну інформацію:   * 1. Перелік сімей з найбільшою кількістю мешканців району Х;   2. Перелік сімей з помешканням виду М категорії К.   Х, М , К – вводити з клавіатури |

2 Блок-схема алгоритму рішення задачі.\



3 Текст програми мовою програмування С++.

#include <iostream> *// ввід вивід з консолі та ввід у файли*

#include <string.h> *// робота з рядками substr(), getline, string,*

#include <windows.h> *// SetConsoleOutputCP, SetConsoleCP встановлення кодування*

#include <fstream> *// робота з файлами, ifstream, ofstream, fstream, open(), close()*

using namespace std;

struct family {

*// 0 звичайний*

*// 1 малозабезпечений*

*// 2 чорнобилець*

*// 3 інвалідність*

*// 5 багатодітний*

    int category;

    int num; *// кількість членів сім'ї*

    string area; *// район*

*// 0 особистий дім*

*// 1 багатоповерхівка*

*// 2 барак*

*// 3 без помешкання*

    int homeType;

};

*// створення сім'ї та запис її у файл*

void createFamily() {

    family obj; *// створюємо об'єкт структури family*

    ofstream file; *// створюємо файл під запис*

    string buffer; *// змінна буферу*

    cin.sync();

    cout <<

    "Введіть категорію" <<

    "\n  | 0 звичайний" <<

    "\n  | 1 малозабезпечений" <<

    "\n  | 2 чорнобилець" <<

    "\n  | 3 інвалідність" <<

    "\n  | 5 багатодітний\n" <<

    "Ваш варіант: ";

    cin.sync(); *// очищення буферу*

    cin >> buffer; *// Вводимо значення у буфер*

    cin.sync(); *// очищення буферу*

    obj.category = stoi(buffer); *// перероблюємо значення буферу у int (якщо користувач введе не int то буде помилка )*

*// та записуємо це значення у поле структури category*

*// (нижче за тим самим принципом )*

    cout << "Введіть кількість членів сім'ї: ";

    cin >> buffer;

    cin.sync(); *// очищення буферу*

    obj.num = stoi(buffer);

    cout << "Введіть район (латинськими літерами): ";

    cin >> buffer;

    cin.sync(); *// очищення буферу*

    obj.area = buffer; *// Просто записуємо string*

    cout << "Типи житла\n"

    << "  | 0 особистий дім\n"

    << "  | 1 багатоповерхівка\n"

    << "  | 2 барак\n"

    << "  | 3 без помешкання\n";

    cout << "Введіть тип житла: ";

    cin >> buffer;

    cin.sync(); *// очищення буферу*

    obj.homeType = stoi(buffer);

    file.open("Vasilenkodata.txt", ios::app); *// відкриваємо файл у режимі додавання app*

*// тобто попередні записи видалятись не будуть*

*// приклад запису: {0, 4, "Obolon", 2}*

*// 0 категорія*

*// 4 кількість членів сім'ї*

*// "Obolon" район*

*// 2 тип житла*

    if (file.is\_open()) { *// якщо файл відкритий*

*// записуємо данні зі структури*

        file << "\n{"<< obj.category << ", ";

        file << obj.num << ", ";

        file << obj.area << ", ";

        file << obj.homeType << "}";

        file.close(); *// закриваємо файл*

    }

}

*// Виведення даних сім'ї*

void printFamily(family f)

{

    cout << "Категорія: ";

    switch(f.category) {

        case 0:

            cout << "Звичайна категорія"; break;

        case 1:

            cout << "Малозабезпечена категорія"; break;

        case 2:

            cout << "Чорнобильска категорія"; break;

        case 3:

            cout << "Категорія з інавлідністю"; break;

        case 4:

            cout << "Багатодітна категорія"; break;

    }

    cout << endl << "Кількість членів сім'ї: " << f.num << endl;

    cout << "Район проживання: " << f.area << endl;

    cout << "Тип помешкання: ";

    switch (f.homeType)

    {

    case 0:

        cout <<"Особистий дім "; break;

    case 1:

        cout <<"Багатоповерхівка "; break;

    case 2:

        cout <<"Барак "; break;

    case 3:

        cout <<"Без помешкання "; break;

    }

    cout << endl;

}

*// заповнення файлу структури з рядку з файлу*

family fill(string line) {

*// приклад рядку {1, 4, "obolon", 2}*

    family obj;

    string buffer;

*// беремо перший елемент рядку {1 <-*

    buffer = line[1];

    obj.category = stoi(buffer); *// записуємо значення категорії*

*// беремо 4й елемент рядку {1, 4 <-*

    buffer= line[4];

    obj.num = stoi(buffer); *// записуємо значення кількість членів сім'ї*

*// беремо передостанній елемент рядку*

*//-> 2}*

    buffer = line[line.length()-2];

    obj.homeType = stoi(buffer); *// записуємо значення типу житла*

*// обрізаємо рядок так, аби залишився лише район*

*// {1, 4, " <- ця частина відрізається*

    line = line.substr(8);

*// ця частина відрізається ->", 2}*

    line = line.substr(0, line.length()-5);

*// залишаєтсья лише назва району*

    obj.area = line; *// записуємо її у структуру*

    return obj; *// повертаємо об'єкт струкури family*

}

*// функція 1 Перелік сімей з найбільною кількістю мешканців району Х*

void printByArea(family f[], int fLen)

{

    string area; *// Шуканий район*

    int largestFamily = 0; *// Найбільша кількість членів сім'ї у відповідному районі*

*// Введення шуканого району*

    cin.sync();

    cout << "Введіть район(латинськими літерами):";

    getline(cin, area);

*// Пошук максимальної кількості членів сім'ї у структурах(сім'ях) відповідного району*

    for (int i = 0; i < fLen; i++)

        if (f[i].area == area) *// перевірка чи підходить район*

            if (f[i].num > largestFamily) *// перевірка чи кількість членів сім'ї більша за максимальну*

                largestFamily = f[i].num; *// оновлення максимальної кількількості членів сім'ї*

    for (int i = 0; i < fLen; i++)

        if (f[i].area == area && f[i].num == largestFamily)  { *// якщо район співпадає та кількість членів сім'ї == максимальній*

            cout << "\nCім'я номер "<<i << endl;

            printFamily(f[i]); *// вивід данних сім'ї*

        }

}

*// функція 2 Перелік сімей з помешкання виду Х категорії Х*

void printByHouseAndCategory(family f[], int fLen)

{

    int category, houseType; *// номер категорії та типу домівки для пошуку*

*// Введення шуканих категорії та домівки*

    cout << "Введіть категорію:";

    cin >> category;

    cout << "Введіть тип житла:";

    cin >> houseType;

*// Перебір структур масиву структур f*

    for (int i = 0; i < fLen; i++)

        if (f[i].homeType == houseType && f[i].category == category)  { *// перевірка структури на відповідність шуканим категорії та типу домівки*

            cout << "\nCім'я номер "<<i << endl; *// Виведення номеру сім'ї у масиві структур*

            printFamily(f[i]);*// Вивід даних сім'ї у консоль*

        }

}

family f0[20];

int filesize=0;

void read() {

    string line; *// буфер для запису рядків з файлу*

    ifstream file; *// відкриття файлу під запис*

    filesize=0; *// змінна розміру для кількості рядків у файл*

    file.open("Vasilenkodata.txt", ios::in); *// відкриття файлу під читання*

    if (file.is\_open()) { *// якщо файл відкритий*

        while(getline(file, line))

            filesize++;

        file.close(); *// закриття файлу, бо ми поки рахували його пройшли повністю*

        file.open("Vasilenkodata.txt", ios::in); *// відкриття файлу*

        int i = 0; *// змінна для того аби рухатись по масиву структур f0 (оголошений глобально)*

        do

        {

            f0[i] = fill(line); *// передаємо рядок у функцію яка повертає заповнену структуру*

            i++;

        } while (getline(file, line));

        file.close(); *// закриваємо файл*

    }

}

int main()

{

    SetConsoleCP(65001); *// Встановлення кодування*

    SetConsoleOutputCP(65001); *// Встановлення кодування*

    int option = 0;

    cout <<

       "|1. Вивести сім'ї за типом житла та категорією" << endl

    << "|2. Вивести сім'ї за районом (з найбільшою кількістю членів сім'ї)" << endl

    << "|3. Створити сім'ю" << endl

    << "|4. Вийти з програми" << endl

    << "|Ваша відповідь: ";

    cin >> option;

    switch (option) {

        case 1:

            read();

            printByHouseAndCategory(f0, filesize);

            break;

        case 2:

            read();

            printByArea(f0, filesize);

            break;

        case 3:

            createFamily();

        default:

            break;

    }

    return 0;

}

Данні файлу

{1, 1, "Obolon", 3}

{2, 2, "Obolon", 0}

{3, 4, "Obolon", 2}

{4, 3, "Obolon", 0}

{1, 4, "Obolon", 1}

{0, 4, "Obolon", 2}

{3, 3, "Obolon", 3}

{3, 3, "Obolon", 3}

{3, 1, Obolon, 1}

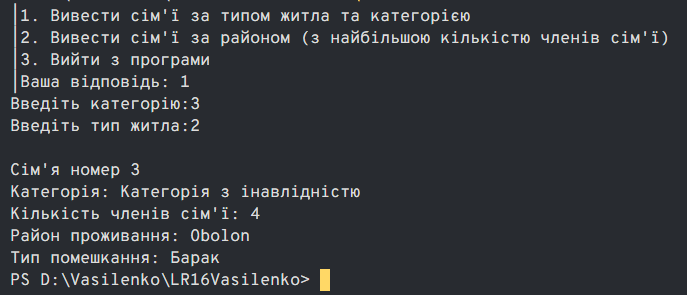
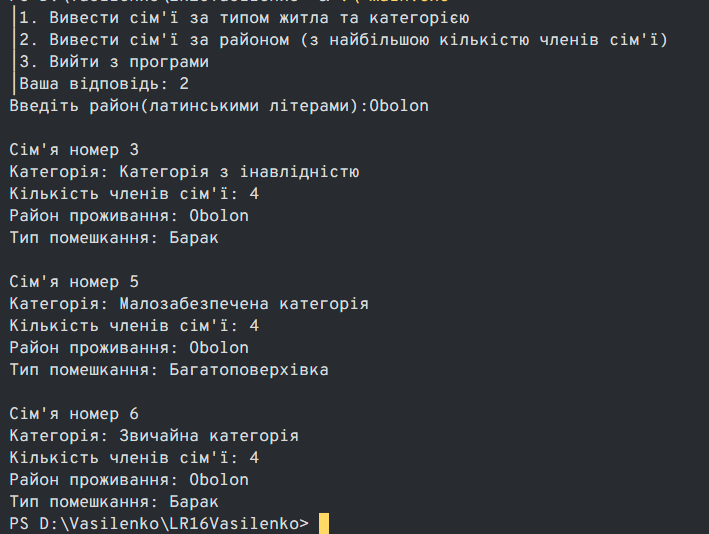
4 Копія вікна виконання програми

Рисунок 1 Виведення сімей за категорією та типом житла



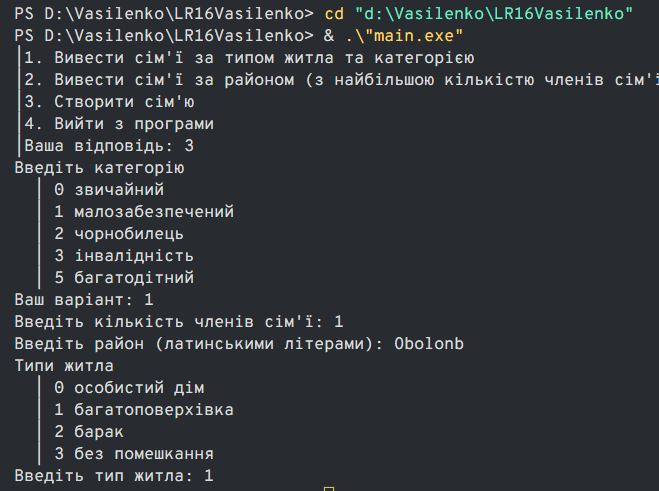
Рисунок 2 Виведення сімей за районом проживання (з найбільшою кількістю членів сім’ї)

Рисунок 3 Створення сім’ї

5 Висновок

В результаті виконання лабораторної роботи опрацьовано теоретичний матеріал за темами:

Обробка структур та файлів

Читання файлів

Запис файлів