Міністерство освіти і науки України

Кіровоградський національний технічний університет

Механіко-технологічний факультет

Кафедра програмування та захисту інформації

Звіт

з виконаної лабораторної роботи № 5

дисципліни “ Модульне програмування ”

на тему

“ Реалізація динамічних структур даних та двійкового введення/виведення ”

Виконав

студент академічної групи КІ-15

Аннаєв А.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Перевірив

старший викладач

Поліщук Л.І.

Кіровоград – 2016

Лабораторна робота №5

**Тема:** Реализация динамических структур данных и двоичного ввода / вывода

**Цель работы:** приобрести практические навыки командного (коллективного) разрабатывания программного обеспечения, использования стандартных средств С / С ++ для работы с динамической памятью и бинарными файловыми потоками при реализации модулей обработки сложных структур данных.

Варіант 5

* Задание на разработку программного обеспечения-

Создать электронный справочник индексов и отделений почтовой связи Украины «УКРПОШТА» (пятизначный индекс, область, район, населенный пункт, название ОПС)

За выбором пользователя (оператора ) программа обеспечивает:

1. Поиск записи в справочнике при вводе индекса или названия населенного пункта;
2. Вывод всего справочника на экран или в текстовый файл;
3. Добавление новой записи в справочник;
4. Извлечение записи с справочника;
5. Сохранение справочника в файл и завершение работы;

Задание 3 : Добавление новой записи в справочник;

Задание 5 : Сохранение справочника в файл и завершение работы;

Текст заголовкового файлу struct\_type\_project\_5.h (з описом елемента динамічної структури):

#ifndef STRUCT\_TYPE\_PROJECT\_5\_H\_INCLUDED

#define STRUCT\_TYPE\_PROJECT\_5\_H\_INCLUDED

struct directory\_elem {

unsigned int index;

char region[20] {'\n'};

char district[20]{'\n'} ;

char city[20] {'\n'};

char post\_office[20] {'\n'};

directory\_elem \*next;

};

#endif // STRUCT\_TYPE\_PROJECT\_5\_H\_INCLUDED

Текст заголовкового файлу Starkina\_unit\_lab\_5.h:

#ifndef STARKINA\_UNIT\_LAB\_5\_H\_INCLUDED

#define STARKINA\_UNIT\_LAB\_5\_H\_INCLUDED

#include <fstream>

#include <cstring>

directory\_elem\* search\_elem(directory\_elem \*head, char city\_in[20]) {

directory\_elem \*p;

p = head;

if (p!=NULL) {

do {

if ((strcmp(p->city,city\_in)==0)) {return p;}

p = p->next;

} while (p!=NULL);

return NULL;

} else {

return p;

}

}

directory\_elem\* search\_elem(directory\_elem \*head, int index\_in) {

directory\_elem \*p;

p = head;

if (p!=NULL) {

do {

if ((p->index == index\_in)) {return p;}

p = p->next;

} while (p!=NULL);

return NULL;

} else {

return p;

}

}

void print\_directoty (directory\_elem \*head, const char name\_file[25] = "\n") {

directory\_elem \*p;

p = head;

if (name\_file == "\n") {

if (p!=NULL) {

do {

std::cout << "Iндекс - " << p->index << std::endl

<< "Область - " << p->region << std::endl

<< "Район - " << p->district << std::endl

<< "Населений пункт - " << p->city << std::endl

<< "ВПЗ - " << p->post\_office << std::endl

<< "-------------------------" << std::endl;

p = p->next;

} while (p!=NULL);}

else {std::cout << "Довiдник порожнiй." << std::endl;}

} else {

std::fstream out\_directory;

out\_directory.open(name\_file, std::ios::out);

if (p!=NULL) {

if (out\_directory) {

do {

out\_directory << "Iндекс - " << p->index << std::endl

<< "Область - " << p->region << std::endl

<< "Район - " << p->district << std::endl

<< "Населений пункт - " << p->city << std::endl

<< "ВПЗ - " << p->post\_office << std::endl

<< "-------------------------" << std::endl;

p = p->next;

} while (p!=NULL);

} else { std::cout << "Помилка!! Файл не вiдкрився." << std::endl;}

} else {out\_directory << "Довiдник порожнiй." << std::endl;}

out\_directory.close();

}

}

#endif // STARKINA\_UNIT\_LAB\_5\_H\_INCLUDED

#include <iostream>

#include <locale>

#include <struct\_type\_project\_5.h>

#include <Starkina\_unit\_lab\_5.h>

#include <Annaev\_unit\_lab\_5.h>

#include <Klimenko\_unit\_lab\_5.h>

using namespace std;

bool take\_directory (directory\_elem \*(&head))

{ directory\_elem \*p;

int index;

fstream data;

data.open("data\_LAB\_5.txt", ios::binary|ios::in);

if (data)

{ if (data.eof()) {return 1;}

else { while (!data.eof())

{ if (head==NULL) {

head = new directory\_elem;

data.read((char\*)&(head->city),sizeof head->city);

data.read((char\*)&(head->index),sizeof head->index);

data.read((char\*)&(head->region),sizeof head->region);

data.read((char\*)&(head->district),sizeof head->district);

data.read((char\*)&(head->post\_office),sizeof head->post\_office);

head->next = NULL;

} else {

p = new directory\_elem;

data.read((char\*)&(p->city),sizeof p->city);

data.read((char\*)&(p->index),sizeof p->index);

data.read((char\*)&(p->region),sizeof p->region);

data.read((char\*)&(p->district),sizeof p->district);

data.read((char\*)&(p->post\_office),sizeof p->post\_office);

p->next = NULL;

index = p->index;

add\_elem\_to\_directory(head,p); } }

delete\_elem\_from\_directory(head,index);

return 1;} }

else {return 0;}

data.close();}

int main()

{

setlocale(LC\_ALL, "ukr");

cout << "Електронний довiдник iндексiв та вiддiлень поштового зв`язку України УДППЗ \"Укрпошта\"" << endl;

char A='Q', Aw;

directory\_elem \*head = NULL;

if (take\_directory(head)) {

do {

cout << "Виведення довiдника ----- 1" << endl

<< "Додавання запису ----- 2" << endl

<< "Вилучення запису ----- 3" << endl

<< "Пошук запису ----- 4" << endl

<< "Збереження довiдника та завершення програми ----- 5" << endl;

cin >> Aw;

switch (Aw) {

case '1': {cout << "Якщо бажаєте зберегти довiдник у файл - введiть A." << endl;

char Ans='Z'; cin >> Ans;

if (Ans=='A') {

char name[25] {'\n'};

cout << "Введiть назву файлу." << endl;

cin >> name;

print\_directoty(head,name);} else { print\_directoty(head);}

break;}

case '2': {int index; char city[20] {'\n'}, region[20]{'\n'}, district[20]{'\n'}, post[20]{'\n'};

cout << "Введiть iндекс:" << endl;

cin >> index;

cout << "Введiть область:" << endl;

cin >> region;

cout << "Введiть район:" << endl;

cin >> district;

cout << "Введiть назву мiста:" << endl;

cin >> city;

cout << "Введiть назву ВПЗ:" << endl;

cin >> post;

add\_elem\_to\_directory(head, index, region, district, city, post);

break;}

case '3': {cout << "Для вилучення запису введiть iндекс" << endl;

int index;

cin >> index;

delete\_elem\_from\_directory(head, index);

break;}

case '4': {cout << "Для пошуку за iндексом введiть -- I, за назвою мiста -- M." << endl;

char S; int index; char city[20] {'\n'};

cin >> S;

if (S=='I')

{cout << "Введiть iндекс:" << endl; cin >> index;

cout << "Iндекс: " << search\_elem(head,index)->index << endl;

cout << "Oбласть: " << search\_elem(head,index)->region << endl;

cout << "Pайон: " << search\_elem(head,index)->district <<endl;

cout << "Мiстo: " << search\_elem(head,index)->city << endl;

cout << "ВПЗ: " << search\_elem(head,index)->post\_office << endl;

}

else {

cout << "Введiть назву мiста:" << endl; cin >> city;

cout << "Iндекс: " << search\_elem(head,city)->index << endl;

cout << "Oбласть: " << search\_elem(head,city)->region << endl;

cout << "Pайон: " << search\_elem(head,city)->district <<endl;

cout << "Мiстo: " << search\_elem(head,city)->city << endl;

cout << "ВПЗ: " << search\_elem(head,city)->post\_office << endl;

}

break;}

case '5': {keep\_directory(head);

A='Y';

break;}

default: cout << "Невизначений символ.\a" << endl;

}

} while (A!='Y');} else {cout << "Виникли проблеми при завантаженi довiдника. Вибачте." << endl;};

return 0;

}

выводы:

При выполнении лабораторной работы приобрел практические навыки командного (коллективного) разрабатывания программного обеспечения, с использования стандартных средств С / С ++ для работы с динамической памятью и бинарными файловыми потоками при реализации модулей обработки сложных структур данных.

Применение динамических структур данных имеет хорошую и плохую строну, для меня использование их затрудняет процесс реализации, но они имеет более гибкую структуру, что позволяет применять их очень часто.

Возникли сложности с бинарными файлами, ведь в бинарный файл нельзя без проблем сохранить и считать кириллицу. Так как считывания из файла и запись происходит побайтно и кириллические буквы имеют разную байтность.