#include <iostream>

#include <fstream>

#include<Windows.h>

#pragma warning (disable:4996)

using namespace std;

struct user

{

char\* fullName;

int IP;

char\* adress;

char\* tariffPlan;

char\* phoneNumber;

};

void addUser(user\*& list, int& size)

{

user\* temp = new user[size + 1];

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

temp[i] = list[i];

delete[] list;

list = temp;

char text[100];

cin.ignore();

cout << "Enter FIO: "; cin.getline(text, 100);

list[size].fullName = new char[strlen(text) + 1];

strcpy(list[size].fullName, text);

cout << "Enter ID: "; cin >> list[size].IP;

cin.ignore();

cout << "Enter adress: "; cin.getline(text, 100);

list[size].adress = new char[strlen(text) + 1];

strcpy(list[size].adress, text);

cout << "Enter tariff plan: "; cin.getline(text, 100);

list[size].tariffPlan = new char[strlen(text) + 1];

strcpy(list[size].tariffPlan, text);

cout << "Enter phone number: "; cin.getline(text, 100);

list[size].phoneNumber = new char[strlen(text) + 1];

strcpy(list[size].phoneNumber, text);

size++;

}

void writeToFile(user\* list, int size)

{

ofstream usr("users.txt");

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

usr << list[i].fullName << "|" << list[i].IP << "|" << list[i].adress << "|" << list[i].tariffPlan << "|" << list[i].phoneNumber << "\n";

usr.close();

}

void readFile(user\*& list, int& size)

{

char buff[200];

ifstream fin("users.txt");

if (!fin)

cout << "Wrong file\n";

else

{

while (fin)

{

fin.getline(buff, 200);

size++;

}

size--;

fin.close();

list = new user[size];

fin.open("users.txt");

char\* data;

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

{

fin.getline(buff, 200);

data = strtok(buff, "|");

list[i].fullName = new char[strlen(data) + 1];

strcpy(list[i].fullName, data);

list[i].IP = atoi(strtok(NULL, "|"));

data = strtok(buff, "|");

list[i].adress = new char[strlen(data) + 1];

strcpy(list[i].adress, data);

data = strtok(buff, "|");

list[i].tariffPlan = new char[strlen(data) + 1];

strcpy(list[i].tariffPlan, data);

data = strtok(buff, "\n");

list[i].phoneNumber = new char[strlen(data) + 1];

strcpy(list[i].phoneNumber, data);

}

fin.close();

}

}

void editData(user\*& list)

{

int idx;

cout << "Choose ur fighter!! "; cin >> idx;

idx--;

char text[100];

cin.ignore();

cout << "Edit user name: "; cin.getline(text, 100);

list[idx].fullName = new char[strlen(text) + 1];

strcpy(list[idx].fullName, text);

cout << "Edit ID: "; cin >> list[idx].IP;

cin.ignore();

cout << "Edit adress: "; cin.getline(text, 100);

list[idx].adress = new char[strlen(text) + 1];

strcpy(list[idx].adress, text);

cout << "Edit tariff plan: "; cin.getline(text, 100);

list[idx].tariffPlan = new char[strlen(text) + 1];

strcpy(list[idx].tariffPlan, text);

cout << "Edit phone number: "; cin.getline(text, 100);

list[idx].phoneNumber = new char[strlen(text) + 1];

strcpy(list[idx].phoneNumber, text);

}

void deleteUser(user\*& list, int& size)

{

user\* temp = new user[size - 1];

int index;

cout << "Enter number in list: "; cin >> index;

index--;

for (int i = 0; i < size - 1; i++)

{

if (i < index)

temp[i] = list[i];

else if (i > index)

temp[i - 1] = list[i];

}

delete[] list[index].fullName;

delete[] list[index].adress;

delete[] list[index].tariffPlan;

delete[] list[index].phoneNumber;

list = temp;

size--;

}

void searchByUserName(user\* list, int size)

{

char userName[100];

cin.ignore();

cout << "Enter user name: "; cin.getline(userName, 100);

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

{

if (strcmp(userName, list[i].fullName) == 0)

cout << i + 1 << "\t" << list[i].fullName << "\t" << list[i].IP << "\t" << list[i].adress << "\t" << list[i].tariffPlan << "\t" << list[i].phoneNumber << "\n";

Sleep(4000);

}

}

void showUsers(user\* list, int size)

{

cout << "#\tUser Name\tIP\t\tAdress\tTariff plan\tPhone number\n";

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

cout << i + 1 << "\t" << list[i].fullName << "\t" << list[i].IP << "\t" << list[i].adress << "\t" << list[i].tariffPlan << "\t" << list[i].phoneNumber << "\n";

}

int main()

{

int size = 0;

user\* list = 0;

int choice;

bool job = 1;

while (job)

{

system("cls");

if (size > 0)

showUsers(list, size);

cout << "enter:\n1 - to add user\n2 - to delete user\n3 - to search by use name\n4 - to write to file\n5 - to read from file\n6 - edit data\n0 - exit\n"; cin >> choice;

switch (choice)

{

case 1:

addUser(list, size);

break;

case 2:

deleteUser(list, size);

break;

case 3:

searchByUserName(list, size);

break;

case 5:

readFile(list, size);

break;

case 4:

writeToFile(list, size);

break;

case 6: editData(list); break;

default:

job = 0;

break;

}

}

}