|  |  |  |
| --- | --- | --- |
| American University of SharjahCollege of Engineering Department of Computer Science & Engineering  P. O. Box 26666  Sharjah, UAE |  | **Lab Instructor:** Eng. Sameer Alawnah  **Office:** EB2-101  **Phone**: 971-6-5152974  **e-mail**: salawnah@aus.edu  **Semester**: Fall 16 |

**CMP 220L – Introduction to Computer Science II**

**Lab 4**

**Note: The good programmer adds comments to his/her code. Add comments to your code.**

**Question 1:**

Create a structure datatype called **Apartment** with the following data members:

* Area: Apartment area in
* Price: Apartment price in Dirhams
* Rooms: Number of rooms
* Bathrooms: Number of bathrooms
* Balconies : Number of balconies

Create the following functions:

* void readApartment(Apartment &apt) ;

This function should read the apartment information from the user.

* void printApartment(Apartment apt);

This function should print the apartment information to the user.

* void changAparmentPrice(Apartment &apt,float newPrice);

This function should change the apartment price

In the main() do the following:

1. Create an instance of the apartment structure (name it *apt*), initialize it based on user input using the readApartment function.
2. Print *apt* using the printApartment function.
3. Change *apts’* price and print it with the new price.

#include <iostream>;

using namespace std;

struct Apartment

{

float area, price;

int rooms, bathrooms, balconies;

};

void readApartment(Apartment &apt)

{

cout << "Area in m^2: ";

cin >> apt.area;

cout << "Apartement price: ";

cin >> apt.price;

cout << "Number of rooms: ";

cin >> apt.rooms;

cout << "Number of bathrooms: ";

cin >> apt.bathrooms;

cout << "Number of balconies: ";

cin >> apt.balconies;

}

void printApartment(Apartment apt)

{

cout << "Apartment Area = " << apt.area << endl;

cout << "Apartment price in Dirhams = " << apt.price<<endl;

cout << "Number of rooms = " << apt.rooms << endl;

cout << "Number of bathrooms = " << apt.bathrooms << endl;

cout << "Number of balconies = " << apt.balconies << endl;

}

void changeApartmentPrice(Apartment &apt, float newPrice)

{

apt.price = newPrice;

cout << "Apartment details after new price: " << endl;

}

void main()

{

Apartment apt;

cout << "Apartment details: " << endl;

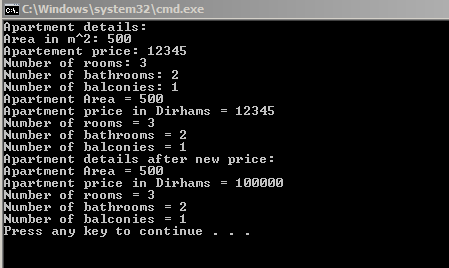
readApartment(apt);

printApartment(apt);

changeApartmentPrice(apt, 100000);

printApartment(apt);

}



**Question 2:**

An input file is used as data source for a telephone directory program, the first line contains the number of contacts, each of the following line represent one contact and consists of the first name, followed by the last name and finally the phone number.

Create a class called “Contact” with the following data and functions members:

* firstName of type string ( private data member).
* lasName of type string (private data member).
* phone of type int (private data member).
* getFirstName() public member function to return the firstName.
* getLastName() public member function to return the lastName.
* getPhone() public member function to return the phone.
* setFirstName(…) public member function to set the firstName.
* setLastName(…) public member function to set the lastName.
* setPhone(…) public member function to set the phone.
* getFullName() public member function to return the full name of the contact.

Write none-member function print(…) that takes a Contact as input and print it to the cout. Your function should make use of the getFullName() function.

Your main() should read the file inputs.txt save it in an array of Contacts and continuously ask the user for the required action from the following actions list:

* 1. Search by First Name.
* 2. Search by Last Name.
* 3. Search by Phone.
* 4. Show the last contact
* 5. Exit.

None: For simplicity, search for the exact string or number in the first 3 actions (use == not find()).

For the first 3 actions, your program should ask the user for the target and print the first name, last name and phone for the found contact, error message otherwise.

In Action 4, you should print the last found contact if exists, error message otherwise

The last action is to exit the whole program.

Good Luck ☺

#include <iostream>

#include <string>

#include<fstream>

using namespace std;

class Contact

{

private:

string firstName;

string lastName;

int Phone;

public:

void setfirstName(string fname){ firstName = fname; }

void setlastName(string lname){ lastName = lname; }

void setPhone(int phone){ Phone = phone; }

string getfirstName(){ return firstName; }

string getlastName(){ return lastName; }

int getPhone(){ return Phone; }

string getFullName()

{

string fullname;

fullname = firstName +" "+ lastName;

return(fullname);

}

};

void print(Contact x)

{

cout << x.getFullName() << " "<<x.getPhone();

}

void main()

{

int size;

ifstream input("phonebook1.txt");

input >> size;

Contact \*contact\_list = new Contact[size];

string fname, lname;

int phone, act;

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

{

input >> fname >> lname >> phone;

contact\_list[j].setfirstName(fname);

contact\_list[j].setlastName(lname);

contact\_list[j].setPhone(phone);

}

while (1)

{

cout << endl<<"1. Search by First Name. " << endl << "2. Search by Last Name. " << endl << "3. Search by Phone." << endl << "4. Show the last contact " << endl << "5. Exit. " << endl;

cout << "Enter required action : ";

cin >> act;

if (act == 1)

{

cout << "Enter name: ";

cin >> fname;

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

{

if (fname == contact\_list[i].getfirstName())

{

print(contact\_list[i]);

break;

}

else

{

cout << "CONTACT NOT FOUND" << endl;

}

}

}

if (act == 2)

{

cout << "Enter name: ";

cin >> lname;

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

if (lname == contact\_list[i].getlastName())

{

print(contact\_list[i]);

break;

}

else

{

cout << "CONTACT NOT FOUND" << endl;

}

}

if (act == 3)

{

cout << "Enter Number: ";

cin >> phone;

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

if (phone == contact\_list[i].getPhone())

{

print(contact\_list[i]);

break;

}

else

{

cout << "CONTACT NOT FOUND" << endl;

}

}

if (act == 4)

{

print(contact\_list[size]);

}

if (act == 5)

{

exit(1);

}

input >> fname >> lname >> phone;

}

}

