**Task 2**

This C++ program is a Contact Management System designed to handle a list of contacts via a console-based interface. It provides users with three core functionalities: adding a new contact, removing an existing contact by its ID, and displaying all saved contacts. Contacts are stored in an array with a maximum capacity defined by `max\_entries`, which is set to 100. Each contact includes an ID, a name, and a phone number. The program offers a menu-driven interface, allowing users to choose an operation by entering the corresponding number. Users can add contacts by entering a name and phone number, delete contacts by specifying the contact ID, and view all stored contacts along with their details. The program continues to run in a loop until the user selects the exit option from the menu.

**Code:**

#include<iostream>

using namespace std;

const int max\_entries = 100;

class CONTACT

{

public:

int id;

string name;

string phonenumber;

public:

CONTACT() {

id = -1;

}

};

class contactmanagementsystem

{

private:

CONTACT contacts[max\_entries];

int contactcount;

public:

contactmanagementsystem()

{

contactcount = 0;

}

void addcontacts()

{

if (contactcount < max\_entries)

{

CONTACT contact;

contact.id = contactcount + 1;

cout << "ENTER NAME OF CONTACT:";

cin.ignore();

getline(cin, contact.name);

cout << "ENTER CONTACT PHONENUMBER:";

getline(cin, contact.phonenumber);contacts[contactcount++] = contact;

}

}

void viewcontacts()

{

cout << "CONTACTS" << endl;

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

{

cout << "ID OF CONTACT IS:" << contacts[i].id << endl;

cout << "NAME OF CONTACT IS:" << contacts[i].name << endl;

cout << "PHONENUMBER OF CONTACT IS:" << contacts[i].phonenumber <<

endl;

}

}

void deletecontacts() {

int id;

cout << "ENTER ID OF CONTACT TO DELETE: ";

cin >> id;

int index = -1;

for (int i = 0; i < contactcount; ++i) {

if (contacts[i].id == id) {

index = i;

}

}

--contactcount;

cout << "CONTACT WITH ID " << id << " DELETED SUCCESSFULLY." << endl;

}

};

int main()

{

contactmanagementsystem cms;

int choice = 0;

do

{

cout << "CONTACT MANAGEMENT SYSTEM" << endl;

cout << "1.ADD CONTACT" << endl;

cout << "2.DELTE CONTACT" << endl;

cout << "3.VIEW CONTACT" << endl;

cout << "enter choice" << endl;

cin >> choice;

switch (choice)

{

case 1:

cms.addcontacts();

break;

case 2:

cms.deletecontacts();

break;

case 3:

cms.viewcontacts();

case 4:

cout << "EXITING..." << endl;

break;

default:

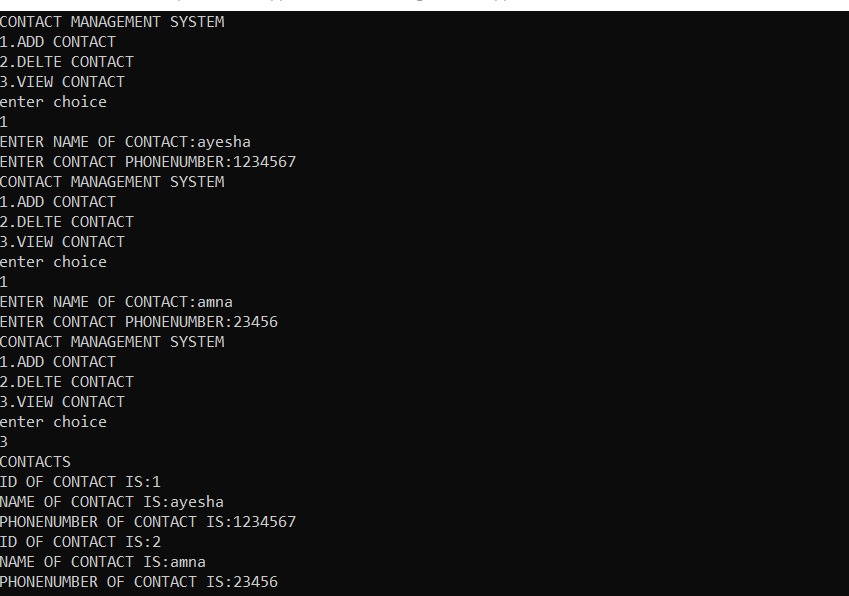
cout << "INVALID CHOICE PLZ ENTER CORRECT CHOICE" << endl;

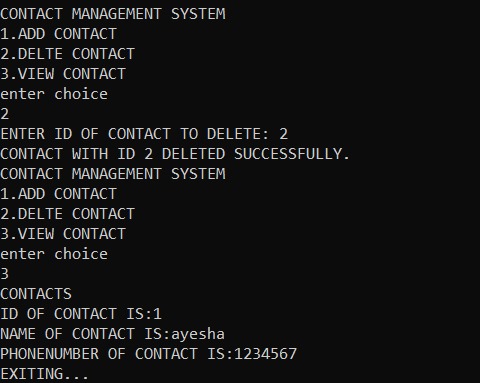
}

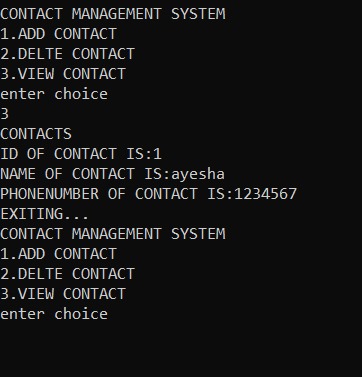
} while (choice != 4);

return 0;

}



****

****