

**COURSE TITLE: Data Structures Lab**

**COURSE CODE: CSE124**

**Project name: Contact Directory**

**SUBMITED TO:**

**MD.AYNUL HASAN NAHID**

Lecturer

Department of CSE

Daffodil International University

**SUBMITED BY:**

**MD.SOHAN MIA**

ID: 0242310005101973

**MD. MUSHFIQ HASSAN CHOWDHURY**

ID: 0242310005101386

**ABIDUL HAK ARMAN**

ID: 0242310005101875

**ASIF ALAM**

ID: 0242310005101970

**Project Overview:**

The objective of this project is to create a simple and user-friendly Contact Management System using C. The program allows users to efficiently manage their contacts by providing features such as adding new contacts, deleting existing contacts, searching for specific contacts, and displaying the entire contact list.

**Tools Required:**

Code::Blocks & Brain

MD.MUSHFIQ HASSAN CHOWDHURY

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

struct Contact

{

    char name[50];

    char phone[20];

    char email[50];

    struct Contact\* next;

};

struct Contact\* head = NULL;

void addContact()

{

    struct Contact\* newContact = (struct Contact\*)malloc(sizeof(struct Contact));

    if (newContact == NULL)

{

        printf("Can not add contact.\n");

        return;

    }

    printf("Enter name: ");

    scanf("%s", newContact->name);

    printf("Enter phone: ");

    scanf("%s", newContact->phone);

    printf("Enter email: ");

    scanf("%s", newContact->email);

    newContact->next = head;

    head = newContact;

    printf("Contact added successfully.\n");

}

MD.SOHAN MIA

void deleteContact()

{

    char searchName[50];

    printf("Enter the name of the contact to delete: ");

    scanf("%s", searchName);

    struct Contact\* current = head;

    struct Contact\* prev = NULL;

    while (current != NULL)

{

        if (strcmp(current->name, searchName) == 0)

{

            if (prev == NULL)

{

                head = current->next;

            }

else

{

                prev->next = current->next;

            }

            free(current);

            printf("Contact deleted successfully.\n");

            return;

        }

        prev = current;

        current = current->next;

    }

    printf("Contact not found.\n");

}

ABIDUL HAK ARMAN

void searchContact() {

    char searchName[50];

    printf("Enter the name to search: ");

    scanf("%s", searchName);

    struct Contact\* current = head;

    while (current != NULL) {

        if (strcmp(current->name, searchName) == 0) {

            printf("Name: %s\n", current->name);

            printf("Phone: %s\n", current->phone);

            printf("Email: %s\n", current->email);

            return;

        }

        current = current->next;

    }

    printf("Contact not found.\n");

}

void displayContacts() {

    struct Contact\* current = head;

    if (current == NULL) {

        printf("Contact list is empty.\n");

        return;

    }

    printf("Contacts:\n");

    while (current != NULL) {

        printf("Name: %s\n", current->name);

        printf("Phone: %s\n", current->phone);

        printf("Email: %s\n", current->email);

        printf("\n");

        current = current->next;

    }

}

ASIF ALAM

void cleanup() {

    struct Contact\* current = head;

    while (current != NULL) {

        struct Contact\* temp = current;

        current = current->next;

        free(temp);

    }

}

int main() {

    int choice;

    while (1) {

        printf("Contact Management System\n");

        printf("1. Add Contact\n");

        printf("2. Delete Contact\n");

        printf("3. Search Contact\n");

        printf("4. Display Contacts\n");

        printf("5. Exit\n");

        printf("Enter your choice: ");

        scanf("%d", &choice);

        switch (choice) {

            case 1:

                addContact();

                break;

            case 2:

                deleteContact();

                break;

            case 3:

                searchContact();

                break;

            case 4:

                displayContacts();

                break;

            case 5:

                cleanup();

                printf("Exiting the program.\n");

                return 0;

            default:

                printf("Invalid choice. Please try again.\n");

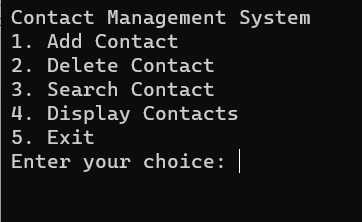
        }

    }

    return 0;

}

**OUTPUT:**

****

**Conclusion:**

The Contact Management System project not only serves as a practical tool for organizing contacts but also provides an educational experience in C programming. It demonstrates the implementation of key concepts while addressing a real-world need for efficient contact management. The simplicity of the system makes it accessible to users with varying levels of technical expertise.