

Project

Submitted To Sir Usman Ghani

Submitted By Safwan Rao

Roll No S23BDOCS1M01225

Subject Data structure & Algorithm

Department Computer Science

\* @file phone\_directory.cpp

\* @brief A simple phone directory application to add, search, and display contacts.

#include <iostream>

#include <vector>

#include <string>

using namespace std;

\* @struct Contact

\* @brief Structure to hold contact information.

struct Contact {

string name; ///< Name of the contact

string phoneNumber; ///< Phone number of the contact

};

/\*\*

\* @brief Adds a new contact to the contacts vector.

\*

\* Prompts the user to enter the name and phone number of a new contact,

\* and adds the contact to the contacts vector.

\*

\* @param contacts Vector to store contact information.

\*/

void addContact(vector<Contact>& contacts) {

Contact newContact;

cout << "Enter name: ";

cin >> newContact.name;

cout << "Enter phone number: ";

cin >> newContact.phoneNumber;

contacts.push\_back(newContact);

}

/\*\*

\* @brief Searches for a contact by name in the contacts vector.

\*

\* Searches the contacts vector for a contact with the specified name.

\* If found, it prints the contact's information; otherwise, it prints

\* a "Contact not found" message.

\*

\* @param contacts Vector storing contact information.

\* @param name Name of the contact to search for.

\*/

void searchByName(const vector<Contact>& contacts, const string& name) {

for (const auto& contact : contacts) {

if (contact.name == name) {

cout << "Found contact: " << contact.name << " (" << contact.phoneNumber << ")\n";

return;

}

}

cout << "Contact not found.\n";

}

/\*\*

\* @brief Displays all contacts in the contacts vector.

\*

\* Prints the name and phone number of each contact stored in the contacts vector.

\*

\* @param contacts Vector storing contact information.

\*/

void displayContacts(const vector<Contact>& contacts) {

cout << "Contacts:\n";

for (const auto& contact : contacts) {

cout << contact.name << " (" << contact.phoneNumber << ")\n";

}

}

/\*\*

\* @brief Main function of the phone directory application.

\*

\* Provides a menu-driven interface for the user to add contacts, search by name,

\* display all contacts, or exit the application.

\*

\* @return int Exit status of the program.

\*/

int main() {

vector<Contact> contacts; // Store contacts

int choice;

do {

cout << "\nPhone Directory Application\n";

cout << "1. Add Contact\n";

cout << "2. Search by Name\n";

cout << "3. Display All Contacts\n";

cout << "4. Exit\n";

cout << "Enter your choice: ";

cin >> choice;

switch (choice) {

case 1:

addContact(contacts);

break;

case 2:

cout << "Enter name to search: ";

string searchName;

cin >> searchName;

searchByName(contacts, searchName);

break;

case 3:

displayContacts(contacts);

break;

case 4:

cout << "Exiting. Goodbye!\n";

break;

default:

cout << "Invalid choice. Try again.\n";

}

} while (choice != 4);

return 0;

}