

TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI TP HCM

KHOA CÔNG NGHỆ THÔNG TIN

ỨNG DỤNG DANH BẠ ĐIỆN THOẠI

Giảng viên hướng dẫn: Trần Thị Mỹ Tiên

Sinh viên thực hiện: Liêu Vĩnh Toàn

Lớp: CN2405CLCD

MỤC LỤC

I. Giới thiệu

II. Cấu trúc dự án

III. Chi tiết nội dung từng file

IV. Kết luận

I. GIỚI THIỆU

Dự án "Ứng dụng danh bạ điện thoại" được thực hiện với mục tiêu xây dựng một hệ thống quản lý danh bạ đơn giản nhưng hiệu quả, giúp lưu trữ, tìm kiếm, chỉnh sửa và xóa các liên hệ một cách dễ dàng. Chương trình được viết bằng ngôn ngữ lập trình C++ và sử dụng tệp tin để lưu trữ dữ liệu.

II. CẤU TRÚC DỰ ÁN

Dự án gồm 3 file chính:

- Data.txt

- main.cpp

- Readme.md

III. CHI TIẾT NỘI DUNG TỪNG FILE

1. File Data.txt

- Đây là tệp chứa toàn bộ thông tin liên lạc, có thể được thêm vào từ giao diện console hoặc chỉnh sửa trực tiếp trong file.

- Cú pháp nhập dữ liệu:

Tên liên lạc, Số điện thoại, Mối quan hệ (1-5)

(các phần cách nhau bằng dấu phẩy).

- Danh mục mối quan hệ:

1. Gia đình
2. Bạn bè
3. Thầy/Cô
4. Đồng nghiệp
5. Khác

- Mỗi dòng trong file chứa một liên hệ duy nhất, không để trống dòng.

- Khi chương trình khởi động, dữ liệu trong file sẽ được đọc vào bộ nhớ (vector).

2. File main.cpp

2.1. CẤU TRÚC:

Chia làm 3 phần:

2.1.1. Cấu trúc dữ liệu:

- struct Contacts: Lưu thông tin mỗi liên hệ gồm:

name: Tên liên hệ (string)

number: Số điện thoại (string)

relation: Mối quan hệ (int)

- vector<Contacts> storeContacts: Danh sách chứa toàn bộ liên hệ.

- ifstream và ofstream được sử dụng để đọc và ghi dữ liệu từ file Data.txt.

2.1.2. Các tính năng chính gồm sáu hàm:

- showAllContacts(): Hiển thị toàn bộ danh bạ.
- addNewContact(): Thêm liên hệ mới vào vector và cập nhật file.
- searchContact(): Tìm kiếm liên hệ theo tên.
- changeInformation(): Sửa thông tin liên hệ.
- deleteContact(): Xóa một liên hệ khỏi danh bạ.
- sortByRelation(): Hiển thị danh bạ theo từng nhóm quan hệ.

2. 1. 3. Các hàm hỗ trợ:

- toLowerCase(): Chuyển chuỗi thành chữ thường.
- checkDuplicate(): Kiểm tra trùng lặp tên hoặc số điện thoại.
- updateData(): Đọc dữ liệu từ file vào vector.
- Update(): Cập nhật dữ liệu sau chỉnh sửa hoặc xóa.
- showUI(): Hiển thị menu ứng dụng.
- Processing(): Nhận lựa chọn và gọi các hàm xử lý.
- setRelation(): Hỗ trợ nhập mối quan hệ hợp lệ.

2. 2. CHI TIẾT TỪNG HÀM

2. 2. 1. Các hàm chức năng chính

2. 2. 1. 1 showAllContacts()

- Kiểm tra danh bạ có rỗng không bằng empty().
- Nếu không rỗng, dùng mảng string Relation[6] để ánh xạ số thành loại quan hệ.
- Duyệt toàn bộ danh sách bằng vòng for để hiển thị thông tin.
- Hiển thị thông tin dưới dạng bảng với từng liên hệ cách nhau bằng dấu phân cách.

2. 2. 1. 2 addNewContact()

- Kiểm tra trùng lặp tên và số điện thoại bằng checkDuplicate().
- Thêm thông tin vào vector storeContacts bằng push_back().
- Mở file Data.txt bằng ofstream (ios::app) để ghi thêm liên hệ.
- Đảm bảo dữ liệu trong file không bị ghi đè bằng cách sử dụng chế độ ghi tiếp (append).

2. 2. 1. 3 searchContact()

- Tìm kiếm liên hệ theo tên bằng cách duyệt danh bạ.
- Sử dụng toLowerCase() để so sánh không phân biệt hoa/thường.

- Xóa khoảng trắng trong tên (erase(remove(...), ...)) trước khi so sánh.
- Nếu tìm thấy liên hệ, hiển thị thông tin chi tiết, nếu không, thông báo "Không tìm thấy".

2.2.1.4 changeInformation()

- Nhận số điện thoại cần thay đổi.
- Kiểm tra xem số điện thoại có tồn tại không.
- Nếu có, nhận thông tin mới và cập nhật vào file.
- Sử dụng ifstream để đọc dữ liệu hiện có và ofstream với ios::trunc để ghi lại dữ liệu mới.

2.2.1.5 deleteContact()

- Xóa một liên hệ khỏi danh bạ bằng cách duyệt danh sách và loại bỏ liên hệ có tên trùng khớp.
- Đọc file (ifstream), lọc bỏ liên hệ bị xóa, ghi lại file (ofstream).
- Hiển thị thông báo xác nhận sau khi xóa thành công.

2.2.1.6 sortByRelation()

- Duyệt danh bạ và lọc các liên hệ theo mối quan hệ đã chọn.
- Hiển thị danh sách các liên hệ thuộc nhóm quan hệ tương ứng.
- Nếu danh sách trống, thông báo "Không có liên hệ nào thuộc nhóm này".

2.2.2. Các hàm hỗ trợ

2.2.2.1 toLowerCase()

- Duyệt từng ký tự trong chuỗi và chuyển về chữ thường bằng tolower().
- Giúp việc so sánh tên không phân biệt hoa/thường.

2.2.2.2 checkDuplicate()

- Kiểm tra xem tên hoặc số điện thoại đã tồn tại chưa.
- Duyệt danh sách, loại bỏ khoảng trắng, so sánh không phân biệt hoa/thường.

2.2.2.3 updateData()

- Đọc dữ liệu từ file Data.txt khi chương trình khởi động.
- Sử dụng ifstream và stringstream để tách dữ liệu thành name, number, relation.
- Lưu danh sách liên hệ vào vector<Contacts> storeContacts.

2.2.2.4 Update()

- Cập nhật dữ liệu sau khi chỉnh sửa hoặc xóa liên hệ.
- Đọc lại toàn bộ dữ liệu từ file Data.txt và làm mới storeContacts.

2.2.2.5 showUI()

- Hiển thị menu điều hướng chính cho người dùng.
- Cung cấp các tùy chọn thao tác với danh bạ.

2.2.2.6 Processing()

- Nhận lựa chọn từ người dùng và gọi hàm xử lý tương ứng bằng switch-case.
- Đảm bảo chỉ nhận giá trị hợp lệ trong phạm vi 1-6.

2. 2. 2. 7 setRelation()

- Hỗ trợ nhập mối quan hệ hợp lệ, chỉ chấp nhận giá trị từ 1 đến 5.
- Nếu nhập sai, yêu cầu người dùng nhập lại.

IV. KẾT LUẬN

Dự án "Ứng dụng danh bạ điện thoại" cung cấp một hệ thống quản lý danh bạ hiệu quả với các tính năng cơ bản như thêm, xóa, tìm kiếm, chỉnh sửa và phân loại liên hệ. Chương trình sử dụng C++ và khai thác các tính năng như xử lý file (ifstream, ofstream), xử lý chuỗi, cấu trúc dữ liệu vector, vòng lặp và câu lệnh điều kiện.

Ngoài ra, việc kiểm tra trùng lặp, xử lý đầu vào và hiển thị dữ liệu một cách có hệ thống giúp chương trình dễ sử dụng và trực quan hơn. Trong tương lai, có thể phát triển thêm giao diện đồ họa hoặc tích hợp cơ sở dữ liệu để lưu trữ dữ liệu lâu dài.

Em xin chân thành cảm ơn cô đã đọc bài giới thiệu của em!