

**ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN**

**KHOA: CÔNG NGHỆ THÔNG TIN**

-----🙞🙜🕮🙞🙜-----



**ĐỒ ÁN CUỐI KỲ**

**ĐỀ TÀI: QUẢN LÍ THƯ VIỆN**

**Giáo viên hướng dẫn: Võ Hoài Việt**

Lớp: 18CTT2

Sinh viên thực hiện: Nguyễn Văn Thìn - 1712787

Nguyễn Hoàng Ngọc Phú - 1712663

Thủ Đức, ngày 24 tháng 5 năm 2019

MỤC LỤC:

1. GIỚI THIỆU……………………………………. 3
2. PHÂN TÍCH……………….……………………..3
3. THIẾT KẾ………………………………………..4
4. PHÂN CÔNG CÔNG VIỆC……………………..9
5. PHƯƠNG PHÁP…………………………………9
6. KẾT QUẢ………………………………………...9- 10

**I. GIỚI THIỆU:**

Giới thiệu để tài đồ án:

-Thư viện trường Đại học Khoa học tự nhiên TP.HCM cần viết một chương trình quản lí thư viện trên nền console, cần quản lí 4 loại thông tin gồm: người dùng, độc giả, sách và các phiếu mượn/trả sách.

* Thông tin người dùng: Tên đăng nhập, mật khẩu, họ tên, ngày sinh, CMND, Địa chỉ, Giới tính, Tình trạng (activated hoặc block). Có 2 loại người dùng là quản lý thư viện và chuyên viên.
* Thông tin thẻ độc giả cần quản lí bao gồm: mã độc giả, họ tên, CMND, ngày tháng năm sinh, giới tính, email, địa chỉ, ngày lập thẻ và ngày hết hạn của thẻ (48 tháng kể từ ngày lập thẻ).
* Thông tin sách cần quản lí bao gồm: ISBN (mã sách), tên sách, tác giả, nhà xuất bản, năm xuất bản, thể loại, giá sách, số quyển sách.
* Mỗi phiếu mượn/trả sách chứa thông tin về mã độc giả, ngày mượn, ngày trả dự kiến, ngày trả thực tế và danh sách ISBN của các sách được mượn. Mỗi sách được mượn tối đa trong 7 ngày, nếu quá hạn sẽ bị phạt tiền 5.000 đồng/ngày. Nếu sách bị mất thì độc giả đó sẽ bị phạt số tiền tương ứng 200% giá sách.

Với 1 số lượng sách lớn trong thư viện thì phần mềm này khá hữu ích, giúp các nhân viên quản lí thư viện quản lí được một cách dễ dàng hơn.

**II. PHÂN TÍCH:**

Để quản lí được một số lượng lớn thông tin như trên thì ta sử dụng struct để lưu những thông tin có liên quan lại với nhau để dễ xử lí.

Tạo struct chứa thông tin người dùng, struct chứa thông tin của độc giả và struct chứa thông tin của sách. Ở mỗi struct này lại có rất nhiều thông tin bên trong, do đó ta sử dụng danh sách liên kết để tạo liên kết riêng cho thông tin của mỗi struct. Nhờ đó ta có thể dễ dàng quản lí những chuỗi thông tin chứa trong mỗi struct này, do nó đã được liên kết với nhau, và cuối cùng ta sẽ lưu giữ tất cả những thông tin nà lại bằng cách ghi vào file. Chương trình này sẽ cho phép chúng ta thực hiện những chức năng:

* Đầu tiên là đăng nhập: Chỉ có duy nhất một tài khoản admin, tài khoản này có quyền cao nhất trong chương trình, nó cho phép người đăng nhập tạo các tài khoản chuyên viên hay quản lí(dùng quản lí thư viện, có cấp bậc thấp hơn so với admin), tài khoản admin này cũng thực hiện được những chỉnh sửa cơ bản
* Sau khi đăng nhập, người dùng sẽ thực hiện chức năng quản lí độc giả như tạo, xóa, tìm kiếm, chỉnh sửa thông tin cũng như xem danh sách độc giả hiện có trong thư viện
* Tiếp đến là quản lí sách, chức năng này tương tự như quản lí độc giả, ta cũng sẽ thêm, xóa nếu sách bị mất, tìm kiếm, và xem số sách đang có ở thư viện
* Ở chức năng tiếp theo, ta sẽ có những thống kê cơ bản trong thư viện về số lượng độc giả, tỷ lệ giới tính, số lượng sách, số lượng sách theo thể loại hiện đang có trong thư viện,…
* Chức năng cuối cùng mà ta cần có ở phần mềm này là tạo phiếu mượn/ trả sách, để thực hiện được việc mượn trả sách thì ta cần có mã độc giả, ngày mượn, ngày trả, isbn cũng số tiền phạt cần phải trả nếu trả quá thời hạn hay bị mất sách.

**III. THIẾT KẾ:**

**+ Chức năng 1:**

Để lưu trữ những thông tin trong chương trình, nhóm đã quyết định sử dụng kiểu dữ liệu cấu trúc để quản lí các nhóm thông tin:

Khai báo struct **USER** và 1 struct **INFOUSER** để lưu trữ thông tin người dùng:

struct USER

{

char username[50];

char password[50];

int kind, local;

};

struct INFOUSER

{

char nameUser[50];

DATE birthday;

char idCard[15];

char address[50];

int gender;

int ActiveOrBlock;

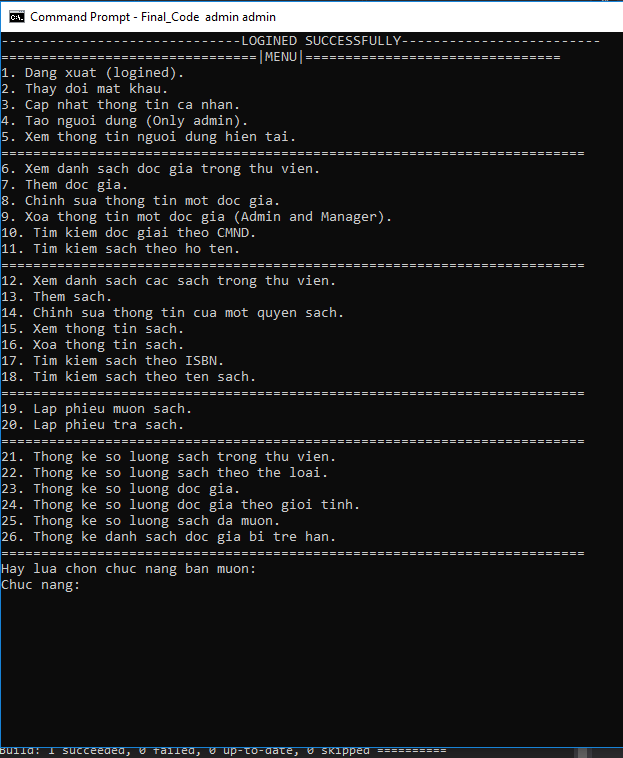
int userkind;

};

Ở đây **kind** là biến kiểu số nguyên, ta sẽ nhập vào lựa chọn 0 hoặc 1( ứng với 0 thì tài khoản này sẽ thuộc kiểu quản lí, ứng với 1 là chuyên viên).

Sau đó, ta ghi tất cả những thông tin này vào file theo kiểu nhị phân( không ghi theo kiểu văn bản để bảo mật thông tin của người dùng cũng như tránh trường hợp có người tùy ý thay đổi thông tin trong file).

Để thực hiện những yêu cầu ở chức năng 1, trước tiên là đăng nhập, phần này ta sẽ dùng command line, ta nhập vào 2 tham số tương ứng với tài khoản và mật khẩu của admin( cũng có thể là chuyên viên hay quản lí), ta so sánh các chuỗi được nhập với thông tin về tài khoản và mật khẩu đã được ghi trong file, nếu trùng khớp thì đăng nhập thành công, ngược lại đăng nhập thất bại.



Đối với phần tạo thêm người dùng thì ta chỉ việc nhập thông tin rồi ghi thông tin vào cuối file. Admin là người có chức năng cao nhất, admin được phép phân loại người dùng thuộc kiểu chuyên viên hay quản lí, cũng như được phép cấp phát tình trạng active hay block. Trong trường hợp quản lí hay chuyên viên bị block sẽ không được phép đăng nhập để thực hiện các chức năng của mình.

Phần cập nhật thông tin cá nhân: trước tiên ta đọc thông tin của danh sách người dùng chứa trong file và đưa chúng ra 1 biến thay đổi sau đó lại ghi vào file( phần thay đổi mật khẩu ta cũng dùng phương pháp này).

**+ Chức năng 2:**

Trước tiên ta cũng dùng struct lưu thông tin độc giả.

struct Reader

{

char ID[12];

char Name[50];

char NationID[12];

DATE Birthday;

int Gender;

char Email[50];

char Address[50];

DATE CreateDay;//ngay lam the

DATE ExpDay;//ngay het han

int IsDeleted;

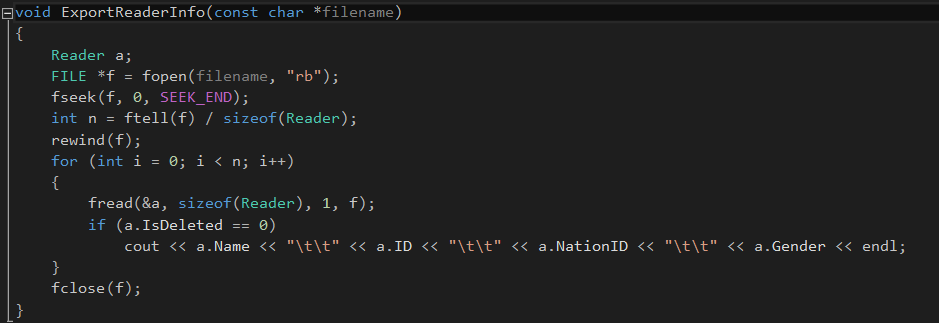
};

Phần chỉnh sửa thông tin và thêm độc giả ta làm tương tự như ở phần người dùng( chức năng 1).

Để xóa thông tin độc giả ta chỉ cần cho trường isDelete = 1.

Phần tiếp theo là tìm kiếm thông tin độc giả theo chứng minh nhân dân, họ tên, ta cũng làm giống phương pháp ở trên duyệt file chứa thông tin độc giả trùng khớp với chứng minh nhân dân hoặc họ tên, cuối cùng là xuất hết thông tin có trong biến duyệt đó ra.

Cuối cùng là xem danh sách các độc giả trong thư viện, ta duyệt toàn bộ file và in ra màn hình.



**+ Chức năng 3:**

Ở phần chức năng 3 này cách làm tương tự với chức năng 2 ở trên.

Tạo struct cái này khá giống chức năng 1 chia làm 2 struct.

struct BOOK

{

char ISBN[15];

char NameB[30];

int local;

int isDeleted;

int cur;

};

struct INFOBOOK

{

char Author[30];

char publisher[30];

int publishingYear;

int number;

int price;

char kindB[30];

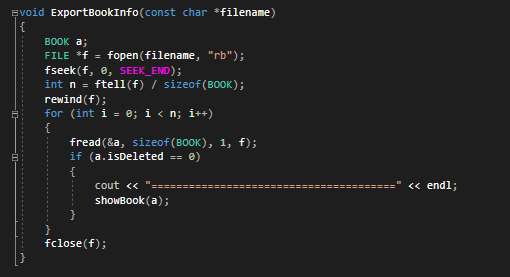
};

Thêm sách bằng cách nhập thông tin cơ bản vào file này sau đó nhập thông tin chi tiết vào file khác kết nối với nhau bằng trường local:

Chỉnh sửa thông tin sách trong biến duyệt và ghi lại toàn bộ vào file

Xóa thông tin sách bằng cách chuyển isDeleted = 1;

Và cuối cùng là xem danh sách các sách có trong thư viện



**+ Chức năng 4:**

Để mượn sách trong thư viện đầu tiên ta cũng tạo một struct để lưu thông tin của người mượn, sách, ngày mượn/ trả,….

struct BorrowCard

{

char IDReader[50];

DATE borrowDay; // ngay muon

DATE returnDay1; // ngay tra du kien

DATE returnDay2; // ngay tra thuc te

int N; // so sach muon

int local;

int IsLate; // 0 neu ko tre; 1 neu tre

int Return = -1; // 0 neu chua tra; 1 neu tra roi

};

Thiết lập hàm mượn sách: Nếu bạn đang mượn một vài cuốn sách thì điều đầu tiên thì bạn phải trả lại những cuốn sách cũ đã mượn từ trước rồi sau đó mới được đăng kí để mượn sách mới. Hàm mượn sách này cho phép bạn được phép mượn nhiều cuốn trong 1 lần mượn.N lưu số sách mượn, khi nhập vào mã ISBN một cuốn sách, nó sẽ so sánh xem có sách nào ứng với mã ISBN như vậy không( so sánh với mã ISBN của sách trong struct ở chức năng 3). Nếu có thì mượn thành công và ghi vào file khác. Bên cạnh đó, mỗi khi có sách được mượn, số sách hiện có trong thư viện cũng sẽ bị giảm đi theo số lượng sách mà độc giả đã mượn. Sau đó hiển nhiên ta sẽ ghi thông tin của người mượn vào file.

**+ Chức năng 5:**

Để tạo phiếu trả sách ta chỉ cần để isReturn =1;

Để thực hiện được những yêu cầu này, trước tiên ta chạy vòng lặp for để tìm kiếm xem mã số độc giả đó có mượn sách hay không, nếu có ta tiếp tục chuyển đến phần trả sách, có tổng cộng ba trường hợp ở phần này, trà sách đúng hạn, trả sách trễ hạn và sách bị mất. Nếu trả đúng hạn, ta chỉ việc thu hồi lại sách, số lượng sách trong thư viện sẽ được trả lại về con số ban đầu trước khi độc giả này mượn, đồng thời ghi thông tin của độc giả vào phiếu trả sách để cho việc thống kê sau này. Trường hợp sách bị trả trễ hạn, ngoài việc đưa số lượng số lượng sách về số lượng ban đầu, ghi vào file chứa thông tin những độc giả đã trả sách thì ta cũng phải tính thêm giá tiền mà độc giả này phải trả do trễ hạn. Lúc này ta so sánh thời gian ngày tháng năm trả với ngày tháng năm mượn, đổi tất cả về số ngày để dễ tính toán, nếu quá 7 ngày mượn thì ta lấy số ngày đó nhân với 5000 đồng sẽ thu được số tiền phải trả. Trường hợp mất sách cũng tương tự, nhưng lúc tính giá tiền thì ta chỉ cần lấy giá tiền của sách nhân với 2 thì sẽ thu được số tiền đền bù cho quyển sách bị mất, đồng thời trừ số sách của thư viện đi tương ứng với số sách mà độc giả làm mất.

**+ Chức năng 6:**

Để thực hiện được các tính toán thống kê ở hàm này , ta chỉ cần duyệt thoả điều kiện r cộng vào biến count là xong

Thống kê số lượng sách theo thể loại( nhóm đã thiết kế chương trình là cho người dung tự nhập thể loại sách chứ không phải chọn thể loại các sách có sẵn, vì thế các thể loại sách sẽ trở nên đa dạng hơn).

Như vậy là đã thực hiện xong các chức năng cần có của một chương trình quản lí thư viện, phần kế tiếp là thiết kế bố cục. Phần này nhóm đã dung switch-case , tạo cho người dung các lựa chọn số như 1,2,3,… và cuối cùng đưa hàm ứng với chức năng tương ứng vào chương trình.

Và cuối cùng ta đã được một chương trình quản lí thư viện chạy trên command line tương đối hoàn chỉnh:

**IV. PHÂN CÔNG CÔNG VIỆC:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Chức năng | | | | | |
| 1 | 2 | 3 | 4 | 5 | 6 |
| Nguyễn Văn Thìn | x | x | x | x |  |  |
| Nguyễn Hoàng Ngọc Phú |  |  |  |  | x | x |

Bên cạnh đó việc tổng hợp chương trình, hướng dẫn sử dụng và video demo do

Bạn Thìn phụ trách. Bạn Phú đảm nhiệm tạo các file chứa thông tin để demo chương trình cũng như phụ trách nhiệm vụ làm báo cáo đồ án của nhóm.

V. PHƯƠNG PHÁP:

Chủ yếu là hoạt động nhóm để thảo luận phương hướng, về nhà tự bản thân tự tìm hiểu thêm để đảm nhiệm những nhiệm vụ cần làm của mình, nếu có thắc mắc hay khó khăn sẽ hỗ trợ lẫn nhau thông qua mạng xã hội Facebook để lần họp nhóm sau sẽ đạt được hiệu suất làm việc tốt hơn.

VI. KẾT QUẢ:

- Hoàn thành được chương trình quản lí thư viện chạy trên nền command line theo yêu cầu của đồ án.

- Biết được thêm nhiều kiến thức mới trong quá trình làm đồ án, có kinh nghiệm để thực hiện các đồ án sau này.

- Khả năng làm việc nhóm được củng cố, phát triển.

- Nâng cao khả năng tư duy lập trình, phân tích khi làm việc với ngôn ngữ lập trình C/C++.

----------------------------------KẾT THÚC ĐỒ ÁN---------------------------------

- Đồ án đã hoàn thành nhưng vẫn còn nhiều điểm hạn chế, chưa thực sự hoàn toàn đáp ứng đầy đủ nhu cầu của người sử dụng.

- Đồ án lần này đã đem lại được nhiều kiến thức mới cho nhóm chúng em,

- Kết quả đồ án là sự cố gắng của các thành viên trong nhóm, vì thế nhóm mong nhận được những nhận xét và góp ý từ phía thầy. Nhóm chúng em xin cám ơn.