

TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA CÔNG NGHỆ THÔNG TIN



ĐỒ ÁN LẬP TRÌNH TÍNH TOÁN

TÊN ĐỀ TÀI

**MINH HỌA CHƯƠNG TRÌNH QUẢN LÝ DANH
SÁCH ĐƠN GIẢN TRONG THƯ VIỆN**

Người hướng dẫn: Nguyễn Thị Lệ Quyên

Nhóm sinh viên thực hiện:

Nguyễn Xuân Thịnh	LỚP: 21TCLC_DT3	Nhóm: 5
Trần Hữu Tuân	LỚP: 21TCLC_DT3	Nhóm: 5

Đà Nẵng, 7/2022

LỜI CẢM ƠN

Lời đầu tiên, chúng em xin gửi lời cảm ơn và lòng biết ơn sâu sắc đến cô giáo **TS. Nguyễn Thị Lệ Quyên** (Khoa Công nghệ thông tin – Trường Đại học Bách Khoa Đà Nẵng), người đã giúp chúng em định hình nghiên cứu, tận tình hướng dẫn và chỉ bảo chúng em trong quá trình thực hiện đồ án PBL1 – Đồ án lập trình tính toán.

Chúng em cũng xin gửi lời cảm ơn chân thành tới các bạn Khóa 21 đã ủng hộ khuyến khích trong suốt quá trình thực hiện đồ án.

Do thời gian có hạn, cũng như kinh nghiệm còn thiếu nên trong chương trình cũng như báo cáo này sẽ không tránh khỏi những thiếu sót, hạn chế nhất định. Chúng em rất mong được nhận xét góp ý từ thầy cô.

Em xin chân thành cảm ơn !

Nhóm sinh viên

Trần Hữu Tuân,
Nguyễn Xuân Thịnh

MỤC LỤC

MỤC LỤC	2
DANH MỤC HÌNH VẼ	3
LỜI MỞ ĐẦU	4
BẢNG PHÂN CÔNG NHIỆM VỤ.....	5
1. TỔNG QUAN ĐỀ TÀI	6
2. CƠ SỞ LÝ THUYẾT.....	6
2.1 Ý tưởng	6
2.2 Cơ sở lý thuyết:	6
2.2.1 Cấu trúc dữ liệu danh sách liên kết đơn	6
2.2.2 Cấu trúc dữ liệu con trỏ.....	8
2.2.3 Cấu trúc dữ liệu ngăn xếp và hàng đợi.....	9
3. TỔ CHỨC CẤU TRÚC DỮ LIỆU VÀ THUẬT TOÁN	10
3.1 Phát biểu bài toán : Tìm kiếm sách trong thư viện	10
3.2 Cấu trúc dữ liệu	10
3.3 Thuật toán	10
4. CHƯƠNG TRÌNH VÀ KẾT QUẢ.....	14
4.1 Tổ chức chương trình.....	14
4.2 Ngôn ngữ cài đặt.....	14
4.3 Kết quả	14
4.3.1 Giao diện chính của chương trình	14
4.3.2 Kết quả thực thi của chương trình	15
4.3.3 Nhận xét và đánh giá	16
5. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN	16
5.1 Kết luận.....	16
5.2 Hướng phát triển	17
6. TÀI LIỆU THAM KHẢO	18

7. PHỤ LỤC	18
------------------	----

DANH MỤC HÌNH VẼ

Hình 1: Mô hình danh sách liên kết đơn	7
Hình 2: Minh họa quá trình xử lý của con trỏ	9
Hình 3: Minh họa cấu trúc dữ liệu hàng đợi	10
Hình 4: Sơ đồ khối thể hiện thuật toán thêm 1 cuốn sách.....	11
Hình 5: Sơ đồ khối thể hiện thuật toán tìm kiếm 1 cuốn sách	12
Hình 6: Sơ đồ khối thể hiện thuật toán xóa 1 cuốn sách	13
Hình 7: Giao diện chính của chương trình	15
Hình 8: Giao diện chính của các thông tin về sách	16

LỜI MỞ ĐẦU

Đề tài : Minh họa chương trình quản lý sách đơn giản trong thư viện. Sử dụng cấu trúc dữ liệu danh sách liên kết đơn để cài đặt danh sách chứa nội dung các cuốn sách.

- Mục đích thực hiện đề tài: Xây dựng chương trình quản lý danh sách trong thư viện
- Mục tiêu :
 - + Giải quyết được các chức năng : khởi tạo danh sách, tìm kiếm, thêm hoặc xóa bớt sách, xem và thay đổi nội dung sách.
 - + Xây dựng được chức năng của thư viện để tra cứu tìm kiếm sách dễ dàng hiệu quả hơn.
 - + Chương trình phải đảm bảo được các yêu cầu về lưu trữ ngoài (đọc dữ liệu từ 1 file input vào chương trình, ghi dữ liệu vào file output các cuốn sách hiện có trong thư viện)
- Bài toán : Xây dựng thư viện đơn giản để giúp đọc giả tìm kiếm sách.
- Để giải quyết bài toán ta cần vận dụng cấu trúc dữ liệu danh sách liên kết đơn vào việc cài đặt thư viện.

BẢNG PHÂN CÔNG NHIỆM VỤ

Trần Hữu Tuân	Đọc tài liệu, cài đặt và viết báo cáo, slide về thuật toán xử lý chuỗi sơ đồ cây.
Nguyễn Xuân Thịnh	Đọc tài liệu, cài đặt và viết báo cáo, slide về thuật toán về ngăn xếp, hàng đợi và con trỏ.

1. TỔNG QUAN ĐỀ TÀI

Thư viện trường có nhu cầu quản lý việc sử dụng sách, Thư viện làm các áp phích sách gồm các thông tin sau: mã sách, tên sách, tên tác giả, nhà xuất bản, năm xuất bản, số bản. Sinh viên muốn đăng kí mượn thì tra cứu phích sách rồi ghi vào phiếu yêu cầu mượn. Khi mượn, sinh viên phải sử dụng thẻ thư viện chứa các thông tin về tên, ngày sinh, địa chỉ, lớp, chuyên ngành của người đọc. Mỗi lần mượn sách được ghi nhận bằng phiếu mượn có dạng sau :

Khi sinh viên trả sách, việc trả sách được ghi nhận vào dòng ngày trả, tình trạng trên phiếu mượn và phiếu mượn được lưu để theo dõi.

2. CƠ SỞ LÝ THUYẾT

2.1. Ý tưởng

- Mỗi struct Book sẽ chọn thuộc tính trong struct: xâu để lưu mã số sách, tên sách, tên tác giả, nhà xuất bản, kiểu int để lưu năm mà sách xuất bản và và trạng thái sách.
- Bên cạnh đó còn sử dụng danh sách hạn chế : ngăn xếp và hàng đợi.

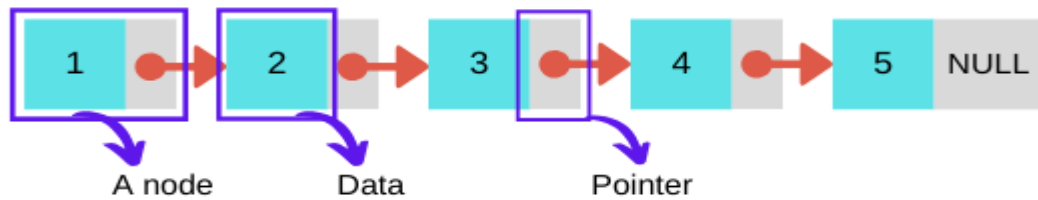
2.2. Cơ sở lý thuyết

2.2.1. Cấu trúc dữ liệu danh sách liên kết đơn:

- Danh sách liên kết đơn (Single Linked List) là một cấu trúc dữ liệu động, nó là một danh sách mà mỗi phần tử đều liên kết với phần tử đứng sau nó trong danh sách. Mỗi phần tử(được gọi là một node hay nút) trong danh sách liên kết đơn là một cấu trúc có 2 thành phần :

- Thành phần dữ liệu : lưu thông tin về bản thân phần tử đó.
- Thành phần liên kết : lưu địa chỉ phần tử đứng sau trong danh sách, nếu phần tử đó là phần tử cuối cùng thì thành phần này bằng NULL.

Single Linked List



Hình 1 : Mô hình danh sách liên kết đơn

- Danh sách liên kết đơn được tạo thành từ nhiều node, do đó, chúng ta sẽ cùng đi từ node trước. Một node gồm hai thành phần là thành phần dữ liệu và thành phần liên kết. Thành phần dữ liệu có thể là kiểu dữ liệu có sẵn hoặc bạn tự định nghĩa (struct hay class...), trong bài viết này để đơn giản mình sẽ sử dụng kiểu int cho phần dữ liệu. Thành phần liên kết là địa chỉ đương nhiên sẽ là con trỏ, con trỏ này trỏ đến node tiếp theo, do đó, con trỏ này là con trỏ trỏ vào một node.

```
struct LinkedList
{
    Node* head;
    Node* tail;
};
```

- Để tạo một node mới, ta thực hiện cấp phát động cho node mới, khởi tạo giá trị ban đầu và trả về địa chỉ của node mới được cấp phát.

```
void CreateList(LinkedList& l)
{
    l.head = NULL;
    l.tail = NULL;
}
```

- Bây giờ để tạo một danh sách, ta làm như sau :

```
LinkedList list;
CreateList(list); // Gán head và tail bằng NULL
```


2.2.2. Cấu trúc dữ liệu con trỏ:

- Con trỏ sử dụng trong cấu trúc là con trỏ cấu trúc trong C, và sử dụng nó để truy cập vào địa chỉ của các thực thể trong bộ nhớ, khai báo cũng như lấy giá trị các thành viên trong nó.

- Dù con trỏ có thể *trở vào* (đại diện cho/là một tên khác của) **nhiều biến khác nhau** trong những thời điểm khác nhau, các đối tượng được trỏ vào phải có **cùng một kiểu dữ liệu**. Kiểu dữ liệu chung này cần được xác định khi khai báo.

- Để khai báo một con trỏ *trở vào* các biến có kiểu dữ liệu `int`, ta bắt buộc khai báo `int *p` hoặc `int* p` (hai cách này như nhau về mặt cú pháp). Tương tự, nếu bạn muốn con trỏ của bạn trỏ vào các biến có kiểu `string` hoặc `queue<char>`, ta phải khai báo `string *s` hoặc `queue<char> *q`.

- Tổng quát về chương trình, nếu `T` là một kiểu dữ liệu nào đó (có thể là kiểu *nguyên thủy* như `int`, `char`, `double` hoặc các struct, class như `vector<string>`, `queue<pair<int, int>>`) thì `T*` là một **con trỏ trỏ vào các biến thuộc kiểu T**. Do con trỏ cũng là một kiểu dữ liệu, con trỏ cũng có thể trỏ vào một con trỏ khác. Ví dụ, `int **p` là một *con trỏ trỏ vào một con trỏ trỏ vào một biến kiểu int* hay `char *****just_to_troll` là một con trỏ trỏ vào một con trỏ trỏ vào một con trỏ trỏ vào một con trỏ trỏ vào một con trỏ trỏ vào một con trỏ trỏ vào một con trỏ trỏ vào một biến kiểu `char`.

- Bản thân con trỏ sẽ được trỏ vào địa chỉ của một biến có cùng kiểu dữ liệu với nó. Cú pháp của phép gán như sau:

```
<ten_con_tro> = &<ten_bien>;
```

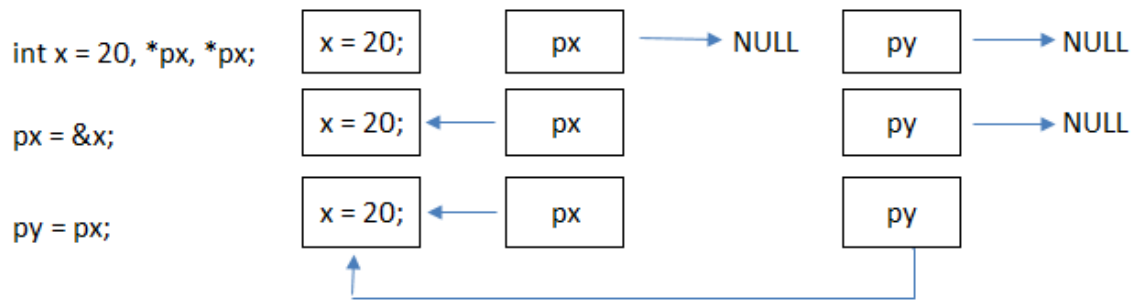
Trong phép toán này, tên con trỏ không có dấu “*”. Ví dụ:

```
int x, *ip;  
ip = &x;
```

Trong phép toán này, phải có dấu con trỏ “*”. Nếu không có dấu con trỏ, sẽ trở thành phép lấy địa chỉ của biến do con trỏ trỏ tới. Ví dụ:

```
int x = 12, y, *ip;  
ip = &y;  
ip = &x;
```

Quá trình trên diễn ra như sau:



Hình 2 : Minh họa quá trình xử lý của con trỏ

2.2.3. Cấu trúc dữ liệu hàng đợi và ngăn xếp:

- Hàng đợi (queue) là một cấu trúc dữ liệu hoạt động theo cơ chế FIFO (First In First Out), tạm dịch là “vào trước ra trước”. Có nghĩa là phần tử nào được thêm hàng đợi trước thì sẽ được lấy ra trước. Có thể hình dung hàng đợi như là một đoàn người xếp hàng mua vé.

- **Các hoạt động cơ bản trên cấu trúc dữ liệu hàng đợi**

Các hoạt động trên cấu trúc dữ liệu hàng đợi để có thể liên quan tới việc khởi tạo hàng đợi, sử dụng dữ liệu trên hàng đợi và sau đó là xóa dữ liệu khỏi bộ nhớ.

Danh sách dưới đây là một số hoạt động cơ bản có thể thực hiện trên cấu trúc dữ liệu hàng đợi:

- **Hoạt động enqueue():** thêm (hay lưu trữ) một phần tử vào trong hàng đợi.
- **Hoạt động dequeue():** xóa một phần tử từ hàng đợi.
- Để sử dụng hàng đợi một cách hiệu quả, chúng ta cũng cần kiểm tra trạng thái của hàng đợi. Để phục vụ cho mục đích này, dưới đây là một số tính năng hỗ trợ khác của hàng đợi:
 - **Phương thức peek() :** lấy phần tử ở đầu hàng đợi, mà không xóa phần tử này.
 - **Phương thức isFull():** kiểm tra xem hàng đợi là đầy hay không.
 - **Phương thức isEmpty():** kiểm tra xem hàng đợi là trống hay hay không.



Hình 3 : Minh họa cấu trúc dữ liệu hàng đợi

3. TỔ CHỨC CẤU TRÚC DỮ LIỆU VÀ THUẬT TOÁN

3.1. Phát biểu bài toán : Tra cứu thông tin của sách thông qua chương trình thư viện

Input: Có 2 dạng : Đọc dữ liệu từ File, nhập từ bàn phím

Output: Cũng có 2 dạng : Hiển thị lên màn hình Console và ghi dữ liệu vào File

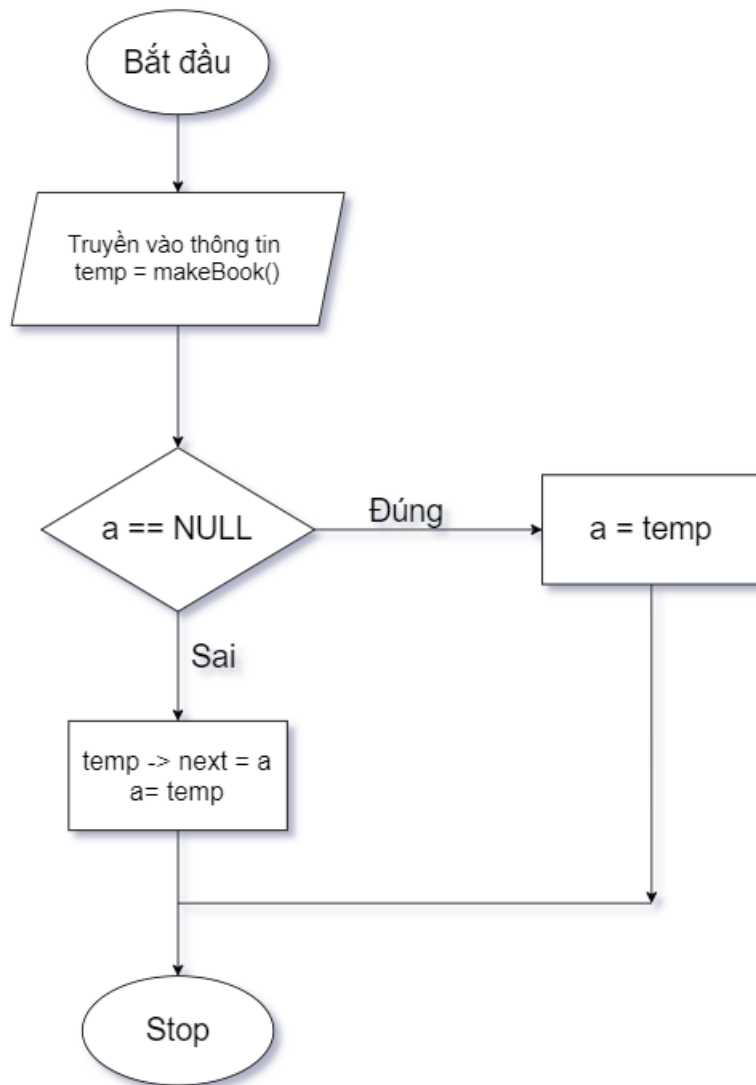
3.2 Cấu trúc dữ liệu

- Sử dụng cấu trúc dữ liệu danh sách liên kết đơn

3.3 Thuật toán

- Trình bày các thuật toán và phân tích độ phức tạp của các thuật toán.

1/ Sơ đồ khối thể hiện thuật toán thêm 1 cuốn sách vào đầu danh sách :



Hình 4 : Sơ đồ khối thể hiện thuật toán thêm 1 cuốn sách vào đầu danh sách

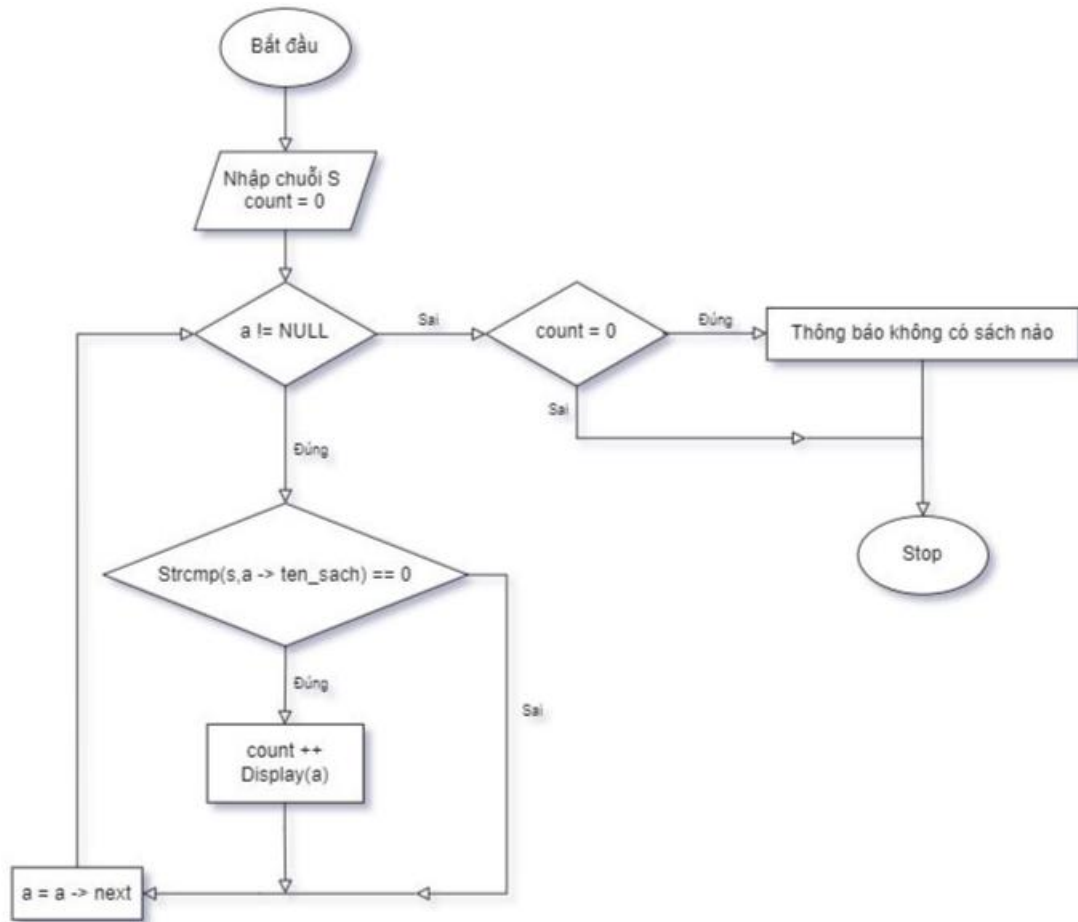
Phân tích độ phức tạp của thuật toán :

+ Việc thêm vào đầu là việc cập nhật thẳng head. Ta gọi Node mới (temp), ta có :

- Nếu head đang trỏ tới NULL, nghĩa là linked list đang trống, Node mới thêm vào sẽ làm head luôn,
 - Ngược lại, ta phải thay thế bằng head cũ bằng head mới
- Việc này phải theo thứ tự như sau :
- Cho temp -> next = a
 - a = temp

Độ phức tạp : $O(1)$

2/ Sơ đồ khối thể hiện thuật toán tìm kiếm 1 cuốn sách :



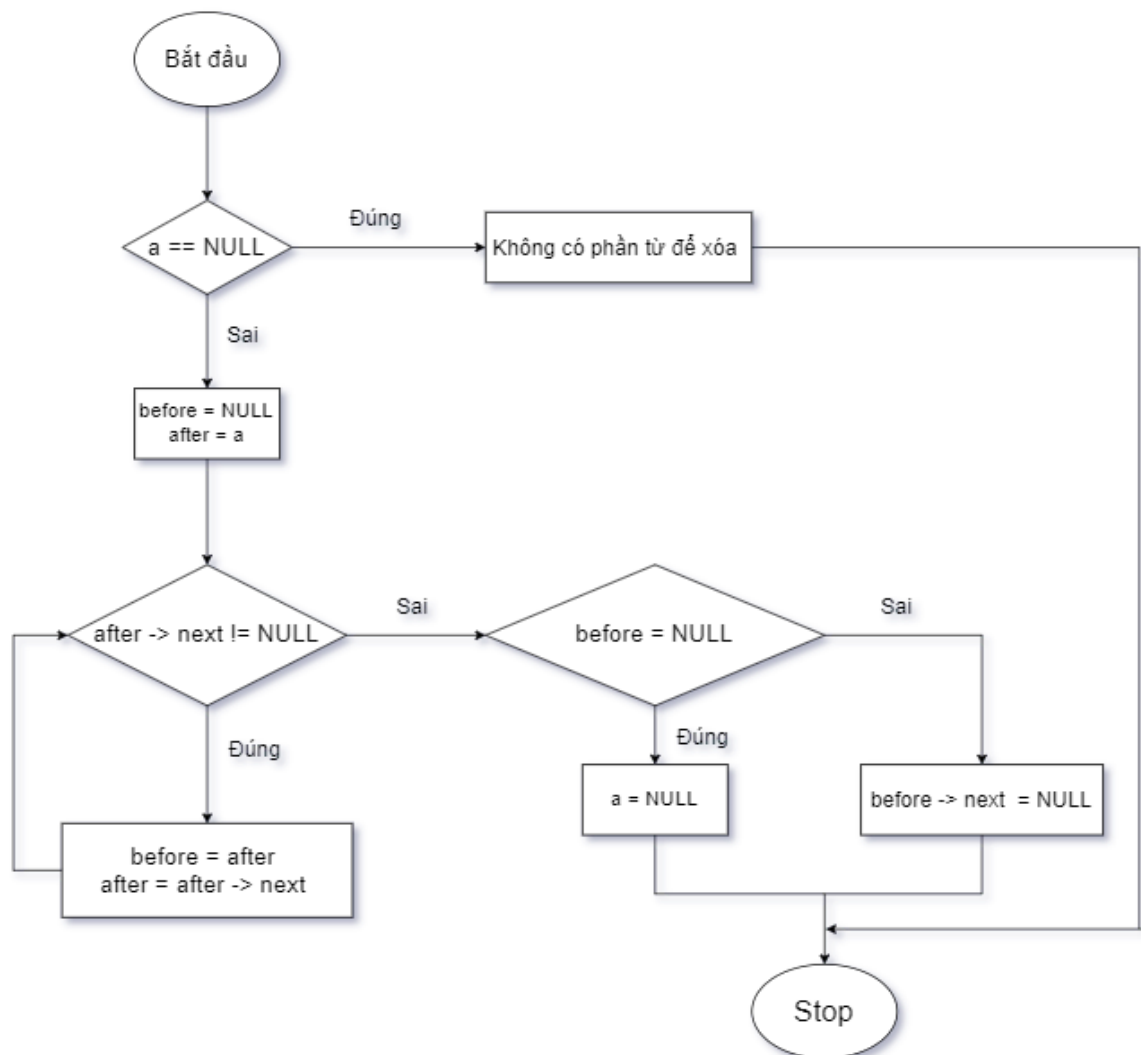
Hình 5 : Sơ đồ khối thể hiện thuật toán tìm kiếm 1 cuốn sách

Phân tích độ phức tạp của thuật toán :

- Điều đầu tiên ta nhập tên sách cần tìm và biến $count = 0$
- Ý tưởng tìm kiếm phần tử cũng là duyệt danh sách, duyệt từng Node cho đến cuối danh sách (NULL)
- Nếu như Node nào có tên sách giống với tên sách cần tìm
($strcmp(s, a \rightarrow ten_sach) == 0$) ta in nó ra màn hình (đồng thời tăng biến đếm $count + 1$)
- Sau khi kết thúc duyệt, nếu $count = 0$ thì ta không tìm thấy cuốn sách cần tìm

Độ phức tạp của thuật toán : $O(n)$ vì duyệt từ đầu đến cuối danh sách

3/ Sơ đồ khối thể hiện thuật toán xóa 1 cuốn sách :



Hình 6 : Sơ đồ khối thể hiện thuật toán xóa 1 cuốn sách

Phân tích độ phức tạp của thuật toán :

- Nếu head đang trỏ tới NULL, nghĩa là linked list đang trống=>Danh sách rỗng không còn gì để xóa.
- Ngược lại ta phải duyệt đến phần tử đó bằng NULL. Ta thực hiện nó như sau :

```
Book before = NULL, after = a;
While (after -> next != NULL)
{
    before = after;
    after = after -> next;
}
If(before == NULL)
{
    a == NULL;
}
else{
    before -> next = NULL;
}
```

Độ phức tạp của thuật toán : $O(n)$.

4. CHƯƠNG TRÌNH VÀ KẾT QUẢ

4.1 Tổ chức chương trình:

Gồm các chức năng sau

- Thêm sách vào thư viện một cách thủ công
- Thêm sách bằng cách đọc file
- Hiển thị và cập nhật thư viện
- Tra cứu
- Lưu sách vào file
- Thay đổi thông tin hoặc xóa sách
- Thoát khỏi chương trình

4.2 Ngôn ngữ cài đặt

- Sử dụng ngôn ngữ lập trình C để cài đặt

4.3 Kết quả

4.3.1 Giao diện chính của chương trình

```
===== MENU =====
1 .Khoi tao danh sach
2 .Them 1 cuon sach
3 .Xoa 1 cuon sach khoi danh sach
4 .Tim kiem sach
5 .Chuc nang xem danh sach
6 .Chuc nang muon/tra sach
7 .Chinh sua noi dung cua sach
8 .Luu file
0 .Ket thuc chuong trinh
=====
Chose=
```

Hình 7 : Giao diện chính của chương trình

=====LIBRARY=====				
Ma so sach	Ten sach	Ten tac gia	Nha xuất bản	Nam sản xuất
0001	Đặc nhân tâm	Dale Carnegie	Simon and Schuster	1936
	CHUA MUON			
0002	The 7 Habits of Highly Effective People	Stephen R. Covey	Free Press	1989
	CHUA MUON			
0003	Tuổi thọ du đôi	Phung Quan	Phung Quan	1988
	DA MUON			
0004	Chiến tranh và hòa bình	Lev Nikolayevich Tolstoy	Ruski Vestnik	1986
	CHUA MUON			
0005	Bố già	Mario Puzo	G. P. Putnam's Sons	1945
	CHUA MUON			
0006	Những người khốn khổ	Victor Hugo	Verboeckhoven & Ce	1862
	DA MUON			
0007	Ông già và biển cả	Ernest Hemingway	Charles Scribner's Sons	1952
	DA MUON			
0008	Điều kỳ diệu của tiệm tạp hóa Namiya	Higashino Keigo	Kadokawa Shoten	2012
	CHUA MUON			
0009	Suối nguồn	Ayn Rand	Bobbs-Merrill Company	1943
	DA MUON			
0010	Catsby vĩ đại	F. Scott Fitzgerald	Charles Scribner's Sons	1925
	CHUA MUON			

Hình 8 : Giao diện chính của các thông tin về sách

4.3.2 Kết quả thực thi của chương trình

Mô tả kết quả thực hiện chương trình.

- Chương trình sẽ đọc dữ liệu từ file text “thuvien.txt” lưu các ký tự Unicode lưu ở ổ đĩa D. Sau các thao tác đọc dữ liệu từ file để thực hiện việc lưu thông các thông tin của struct book. Chương trình yêu cầu là một quyển sách phải có thông tin cụ thể, thao tác với các ký tự trong bảng mã ASCII. Quá trình đọc dữ liệu file text mà các ký tự unicode vào từng thuộc tính của struct sẽ khác với đọc file text mà các ký tự đều có trong bảng mã là ASCII. Sau khi đọc dữ liệu từ file, khi có đầy đủ thuộc tính của 1 struct thì ta phải tiến hành là lưu thông tin vào nơi nào đó để có thể thực hiện các thao tác. Ở đây chúng xem sẽ chọn cấu trúc dữ liệu là danh sách liên kết đơn.
- Việc tiếp theo là cần làm là thêm lần lượt từng biến có kiểu dữ liệu book đã đọc từ file trước đó vào danh sách liên kết. Để thêm vào thì bắt buộc phải có một hàm tối ưu. Hàm này thì có nhiệm vụ xâu eng (xâu lưu chuỗi sách trong struct book).
- Xây dựng hàm insert để thêm 1 biến có kiểu dữ liệu cấu trúc vào danh sách liên kết. Việc thêm 1 phần tử vào danh sách liên kết có thể thực hiện bằng 2 cách đọc dữ liệu từ file hoặc nhập thủ công.
- Thao tác tìm kiếm, xóa một cuốn sách khỏi thư viện thực chất là việc tìm kiếm, xóa xâu eng (một thuộc tính của struct) trong mỗi cuốn sách.

4.3.3 Nhận xét đánh giá

- **Tích cực :**

- Chương trình đã thực hiện tương đối đầy đủ những yêu cầu đề bài đưa ra, đã xây dựng được một thư viện đã mang tính chất khách quan như nhập dữ liệu từ file và thông tin sách từ file đã được tích hợp đầy đủ thông tin.
- Cài đặt dễ dàng ngoại trừ việc đọc ghi file. Vì chương trình chỉ yêu cầu nắm được kiến thức về danh sách liên kết đơn, biết cách sử dụng kiểu dữ liệu char thì việc cài đặt không có gì phức tạp.
- Có chức năng lựa chọn nhiệm vụ thực hiện bằng menu động

- **Hạn chế :**

- Chương trình chưa có được giao diện dễ nhìn như màu sắc, kích thước giữa các dòng sẽ làm khó khăn cho đọc giả khi tìm kiếm.
- Việc in ra bảng menu thì tốc độ in không được nhanh lắm vì cuối mỗi vòng lặp phải in ra lại bảng menu đó, trong khi menu được xây dựng từ nhiều hộp phải in lại tất cả trên hộp đó nên thời gian hơi lâu.
- Nguồn sách chưa được nhiều, vì chủ yếu là thủ công vào file để chương trình đọc dữ liệu và console chưa hiển thị tiếng việt.

5. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

5.1 Kết luận

- Sau một thời gian nghiên cứu, hoàn thiện báo cáo đồ án, chúng em đã thực hiện đầy đủ những yêu cầu mà đề bài nêu ra.
- Đề tài đạt được những mục tiêu đặt ra về mặt lý thuyết cũng như ứng dụng trong thực tiễn.
- Về mặt lý thuyết, đề tài đã trình bày được cơ sở lý thuyết và thuật toán giải quyết bài toán tìm kiếm sách bằng ngăn xếp và hàng đợi.

- Về mặt thực tiễn, đồ án đã bước đầu giải quyết được cách vận hành của một thư viện qua chương trình như thế nào cũng như có thể ứng dụng trong những thư viện chưa được áp dụng.

5.2 Hướng phát triển

- Để chương trình trở nên hoàn thiện hơn, chương trình cần được thiết kế giao diện đẹp mắt hơn, dễ dùng hơn và tối ưu thuật toán hơn với tốc độ nhanh hơn để người dùng đạt được hiệu quả tốt nhất. Qua đó, có thể tạo được một chương trình hỗ trợ học sinh, sinh viên, giảng viên việc giải quyết một cách nhanh chóng và hiệu quả bằng những thuật toán tương tự, cũng như các lĩnh vực có liên quan.

TÀI LIỆU THAM KHẢO

[1] Các thao tác với kiểu char, cách đọc, ghi file với kiểu char

<https://tinyurl.com/3vbmd9mf>

[2] Các hàm khi thao tác với kiểu dữ liệu char

<https://www.geeksforgeeks.org/wide-char-and-library-functions-in-c/>

PHỤ LỤC

```
#include <stdio.h>

#include <stdbool.h>

#include <stdlib.h>

#include <conio.h>

#include <windows.h>

#include <string.h>

typedef char infor[40];
```

```
struct Book
{
    infor ma_so_sach;
    infor ten_sach;
    infor ten_tac_gia;
    infor nha_xuat_ban;
    int nam_xuat_ban;
    int trang_thai_sach;
    struct Book *next;
};

typedef struct Book *book;

book makeBook();

int Size(book a);
book file_data(book a); // 1
book InsertFirst(book a);
book InsertLast(book a);
book InsertAfter(book a);

book Insert_Book(book a); // 2
book DeleteFirst(book a);
book DeleteLast(book a);
book DeleteAfter(book a, int pos);
```

```
book DeleteMaso(book a);  
book DeleteName(book a);  
book DeleteTacGia(book a);  
  
book Delete_Book(book a); //3  
  
void Search_Ten_Sach(book a);  
void Search_Ten_Tac_Gia(book a);  
void Search_Nha_Xuat_Ban(book a);  
void SearchBook(book a); // 4  
  
void Dislay(book a);  
void Xem_Toan_Bo_Sach(book a);  
void Xem_Sach_Da_Muon(book a);  
void Xem_Sach_Chua_Muon(book a);  
  
void swapstring(char *x, char *y);  
void swapint(int *x, int *y);  
  
void Theo_Van_Ten_Sach(book a);  
void Theo_Van_Ten_Tac_Gia(book a);  
void Theo_Van_Nha_Xuat_Ban(book a);  
void Theo_Ngay_Xuat_Ban(book a);  
void Xem_Danh_Sach_Theo_Thu_Tu(book a);  
void Xem_Sach(book a); //5
```

```
book Muon_sach(book a);
book Tra_sach(book a);
void Muon_tra_sach(book a); //6

book Sua_ten_sach(book a);
book Sua_ten_tac_gia(book a);
book Sua_nha_xuat_ban(book a);
book Sua_nam_san_xuat(book a);
void Cap_nhat_noi_dung_cua_sach(book a); // 7

void save_data(book a); // 8

int main()
{
    book head = NULL;
    int chose;
    do
    {
        printf("===== MENU
=====");

        printf("\n1 .Khoi tao danh sach");
        printf("\n2 .Them 1 cuon sach");
        printf("\n3 .Xoa 1 cuon sach khoi danh sach");
        printf("\n4 .Tim kiem sach");
        printf("\n5 .Chuc nang xem danh sach");
```

```
        printf("\n6 .Chuc nang muon/tra sach");
        printf("\n7 .Chinh sua noi dung cua sach");
        printf("\n8 .Luu file");
        printf("\n0 .Ket thuc chuong trinh");

printf("\n=====
=====");

        printf("\n\nChose=");
        scanf("%d", &chose);
        switch (chose)
        {
        case 0:
                break;
        case 1:
                head = file_data(head); // doc du lieu tu file
                system("cls");
                break;
        case 2:
                system("cls");
                head = Insert_Book(head);
                break;
        case 3:
                system("cls");
                head = Delete_Book(head);
                break;
        case 4:
```

```
        system("cls");
        SearchBook(head);
        break;
    case 5:
        system("cls");
        Xem_Sach(head);
        break;
    case 6:
        system("cls");
        Muon_tra_sach(head);
        break;
    case 7:
        system("cls");
        Cap_nhat_noi_dung_cua_sach(head);
        break;
    case 8:
        save_data(head);
        system("cls");
        break;
    default:
        printf("Sai chuc nang, vui long chon lai!\n");
        break;
}
} while (chose);
return 0;
```


}

book makeBook()

{

book temp = (book)malloc(sizeof(struct Book));

temp->next = NULL;

printf("Nhap ma so sach:");

fflush(stdin);

gets(temp->ma_so_sach);

printf("Nhap ten sach:");

fflush(stdin);

gets(temp->ten_sach);

printf("Nhap ten tac gia:");

fflush(stdin);

gets(temp->ten_tac_gia);

printf("Nhap nha xuất bản:");

fflush(stdin);

gets(temp->nha_xuat_ban);

printf("Nhap năm xuất bản:");

scanf("%d", &temp->nam_xuat_ban);

printf("Nhap trạng thái sách(1.Chưa mượn/0.Đã mượn):");

scanf("%d", &temp->trang_thai_sach);

return temp;

}

```
int Size(book a)
{
    int count = 0;
    while (a != NULL)
    {
        ++count;
        a = a->next;
    }
    return count;
}

book file_data(book a)
{
    infor s;
    printf ("Nhap duong dan:");
    fflush(stdin);
    gets(s);
    FILE *f = fopen(s, "r+");
    if(f == NULL)
    {
        printf("Error!");
        exit(1);
    }
    while (1)
    {
```

```
        book temp = (book)malloc(sizeof(struct Book));
        temp->next = NULL;
        char s[256];
        fgets(s, 256, f);
        fscanf(f, "%d %d", &temp->nam_xuat_ban, &temp-
>trang_thai_sach);
        fscanf(f, "\n");
        char *token = strtok(s, ",");

        strcpy(temp->ma_so_sach, token);

        token = strtok(NULL, ",");
        strcpy(temp->ten_sach, token);

        token = strtok(NULL, ",");
        strcpy(temp->ten_tac_gia, token);

        token = strtok(NULL, ",");
        strcpy(temp->nha_xuat_ban, token);
        if (temp->nha_xuat_ban[strlen(temp->nha_xuat_ban) - 1] ==
'\n')
            temp->nha_xuat_ban[strlen(temp->nha_xuat_ban) - 1] = '\0';

        if (a == NULL)
        {
            a = temp;
```

```
    }  
    else  
    {  
        book p = a;  
        while (p->next != NULL)  
        {  
            p = p->next;  
        }  
        p->next = temp;  
    }  
    if (feof(f))  
        break;  
}  
fclose(f);  
return a;  
}
```

```
book InsertFirst(book a)  
{  
    book temp = makeBook();  
    if (a == NULL)  
    {  
        a = temp;  
    }  
    else
```

```
{  
    temp->next = a;  
    a = temp;  
}  
return a;  
}
```

```
book InsertLast(book a)  
{  
    book temp = makeBook();  
    if (a == NULL)  
    {  
        a = temp;  
    }  
    else  
    {  
        book p = a;  
        while (p->next != NULL)  
        {  
            p = p->next;  
        }  
        p->next = temp;  
    }  
    return a;  
}
```

```
book InsertAfter(book a)
{
    int n = Size(a), pos;
    printf("\nNhap vi tri muon chen:");
    scanf("%d", &pos);
    if (pos <= 0 || pos > n + 1)
    {
        printf("Vi tri chen khong hop le!\n");
    }
    if (pos == 1)
    {
        return InsertFirst(a);
    }
    else if (pos == n + 1)
    {
        return InsertLast(a);
    }
    book p = a;
    for (int i = 1; i < pos - 1; i++)
    {
        p = p->next;
    }
    book temp = makeBook();
    temp->next = p->next;
```

```
    p->next = temp;
    return a;
}
```

```
book Insert_Book(book a)
{
    int chose;
    do
    {
        system("cls");
        Xem_Toan_Bo_Sach(a);
        printf("\n=====Them 1 cuon sach=====");
        printf("\n1 .Them vao dau danh sach");
        printf("\n2 .Them vao sau 1 cuon sach nao do");
        printf("\n3 .Them vao cuoi danh sach");
        printf("\n0 .Tro ve");
        printf("\n=====");
        printf("\n\nChose=");
        scanf("%d", &chose);
        switch (chose)
        {
            case 0:
                system("cls");
                break;
            case 1:
```

```
        a = InsertFirst(a);
        break;
    case 2:
        a = InsertAfter(a);
        break;
    case 3:
        a = InsertLast(a);
        break;
    default:
        printf("Sai chuc nang, vui long chon lai!\n");
        break;
    }
} while (chose);
return a;
}
```

```
book DeleteFirst(book a)
{
    if (a == NULL)
    {
        printf("\n=>Khong co phan tu de xoa!");
        return a;
    }
    return a->next;
}
```



```
book DeleteLast(book a)
{
    if (a == NULL)
    {
        printf("\n=>Khong co phan tu de xoa!");
        return a;
    }
    book before = NULL, after = a;
    while (after->next != NULL)
    {
        before = after;
        after = after->next;
    }
    if (before == NULL)
    {
        a = NULL;
    }
    else
    {
        before->next = NULL;
    }
    return a;
}
```

```
book DeleteAfter(book a, int pos)
{
    int n = Size(a);
    if (pos <= 0 || pos > n)
    {
        printf("=>Vi tri muon xoa khong hop le!");
        return a;
    }
    book before = NULL, after = a;
    for (int i = 1; i < pos; i++)
    {
        before = after;
        after = after->next;
    }
    if (before == NULL)
    {
        a = a->next;
    }
    else
    {
        before->next = after->next;
    }
    return a;
}
```

```
book DeleteMaso(book a)
{
    infor s;
    int n = Size(a), pos = 0, i;
    book p = a;
    printf("Nhap ma so sach muon xoa:");
    fflush(stdin);
    gets(s);
    for (i = 1; i <= n; i++)
    {
        if (strcmp(p->ma_so_sach, s) == 0)
            pos = i;
        p = p->next;
    }
    return DeleteAfter(a, pos);
}
```

```
book DeleteName(book a)
{
    infor s;
    int n = Size(a), pos = 0, i;
    book p = a;
    printf("Nhap ten sach muon xoa:");
    fflush(stdin);
    gets(s);
```

```
    for (i = 1; i <= n; i++)
    {
        if (strcmp(p->ten_sach, s) == 0)
            pos = i;
        p = p->next;
    }
    return DeleteAfter(a, pos);
}
```

```
book DeleteTacGia(book a)
{
    infor s;
    int n = Size(a), pos = 0;
    book p = a;
    printf("Nhap ten tac gia muon xoa:");
    fflush(stdin);
    gets(s);
    while (p != NULL)
    {
        pos++;
        if (strcmp(p->ten_tac_gia, s) == 0)
        {
            a = DeleteAfter(a, pos);
            pos--;
        }
    }
}
```

```
        p = p->next;
    }
    return a;
}
```

```
book Delete_Book(book a)
```

```
{
    int chose, pos;
    do
    {
        system("cls");
        Xem_Toan_Bo_Sach(a);
        printf("\n=====Xoa 1 cuon sach=====");
        printf("\n1 .Theo ma so sach");
        printf("\n2 .Theo ten sach");
        printf("\n3 .Theo ten tac gia");
        printf("\n4 .Xoa cuon sach o dau danh sach");
        printf("\n5 .Xoa cuon sach o sau cuon sach co ma so nao do");
        printf("\n6 .Xao cuon sach o cuoi danh sach");
        printf("\n0 .Tro ve");
        printf("\n=====");
        printf("\n\nChose=");
        scanf("%d", &chose);
        switch (chose)
        {
```

```
case 0:
    system("cls");
    break;
case 1:
    a = DeleteMaso(a);
    break;
case 2:
    a = DeleteName(a);
    break;
case 3:
    a = DeleteTacGia(a);
    break;
case 4:
    a = DeleteFirst(a);
    break;
case 5:
    printf("Nhap vi tri muon xoa:");
    scanf("%d", &pos);
    a = DeleteAfter(a, pos);
    break;
case 6:
    a = DeleteLast(a);
    break;
default:
    printf("Sai chuc nang, vui long chon lai!\n");
```

```
        break;
    }
} while (chose);
return a;
}

void Search_Ten_Sach(book a)
{
    infor s;
    int count=0;
    printf("\nNhap ten sach:");
    fflush(stdin);
    gets(s);
    while (a != NULL)
    {
        if (strcmp(s, a->ten_sach) == 0)
        {
            count++;
            Dislay(a);
            if (a->trang_thai_sach == 1)
                printf("CHUA MUON\n");
            else
                printf("DA MUON\n");
        }
        a = a->next;
    }
}
```

```
    }
    if(!count) printf("Sach khong ton tai!\n\n");
}
void Search_Ten_Tac_Gia(book a)
{
    infor s;
    int count=0;
    printf("\nNhap tac gia:");
    fflush(stdin);
    gets(s);
    while (a != NULL)
    {
        if (strcmp(s, a->ten_tac_gia) == 0)
        {
            count ++;
            Dislay(a);
            if (a->trang_thai_sach == 1)
                printf("CHUA MUON\n");
            else
                printf("DA MUON\n");
        }
        a = a->next;
    }
    if(!count) printf("Khong co sach nao ca!\n\n");
}
```



```
void Search_Nha_Xuat_Ban(book a)
{
    infor s;
    int count;
    printf("\nNhap nha xuất ban:");
    fflush(stdin);
    gets(s);
    while (a != NULL)
    {
        if (strcmp(s, a->nha_xuat_ban) == 0)
        {
            count++;
            Dislay(a);
            if (a->trang_thai_sach == 1)
                printf("CHUA MUON\n");
            else
                printf("DA MUON\n");
        }
        a = a->next;
    }
    if(!count) printf("Khong co sach nao ca!\n\n");
}

void SearchBook(book a)
{
```

```
int chose;
do
{
    printf("=====Timkiem sach theo=====");
    printf("\n1 .Ten sach");
    printf("\n2 .Ten tac gia");
    printf("\n3 .Ten nha xuat ban");
    printf("\n0 .Tro ve");
    printf("\n=====");
    printf("\n\nChose=");
    scanf("%d", &chose);
    switch (chose)
    {
        case 0:
            system("cls");
            break;
        case 1:
            system("cls");
            Xem_Toan_Bo_Sach(a);
            Search_Ten_Sach(a);
            break;
        case 2:
            system("cls");
            Xem_Toan_Bo_Sach(a);
            Search_Ten_Tac_Gia(a);
```

```
        break;
    case 3:
        system("cls");
        Xem_Toan_Bo_Sach(a);
        Search_Nha_Xuat_Ban(a);
        break;
    default:
        printf("Sai chuc nang, vui long chon lai!\n");
        break;
    }
} while (chose);
}

void Intro()
{
    printf("\nMa so sach      Ten sach      Ten tac gia
Nha xuất ban      Nam sản xuất      Trang thái sách\n");
}

void Dislay(book a)
{
    printf("%-10s %-40s %-30s %-30s %-20d", a->ma_so_sach, a-
>ten_sach, a->ten_tac_gia, a->nha_xuat_ban, a->nam_xuat_ban);
    if (a->trang_thai_sach == 1)
        printf("CHUA MUON\n");
    else
```

```
        printf("DA MUON\n");
        a = a->next;
    }

void Xem_Toan_Bo_Sach(book a)
{
    printf("=====LIBRARY=====");
    Intro();
    while (a != NULL)
    {
        Dislay(a);

        a = a->next;
    }
}

void Xem_Sach_Da_Muon(book a)
{
    printf("=====LIBRARY=====");
    Intro();
    while (a != NULL)
    {
        if (a->trang_thai_sach == 0)
        {
            Dislay(a);
        }
    }
}
```

```
        }
        a = a->next;
    }
}

void Xem_Sach_Chua_Muon(book a)
{
    printf("=====LIBRARY=====");
    Intro();
    while (a != NULL)
    {
        if (a->trang_thai_sach == 1)
        {
            Dislay(a);
        }
        a = a->next;
    }
}
```

```
void swapstring(char *x, char *y)
{
    infor temp;
    strcpy(temp, x);
```

```
    strcpy(x, y);
    strcpy(y, temp);
}
```

```
void swapint(int *x, int *y)
{
    int temp = *x;
    *x = *y;
    *y = temp;
}
```

```
void Theo_Van_Ten_Sach(book a)
{
    for (book p = a; p->next != NULL; p = p->next)
    {
        book min = p;
        for (book q = p->next; q != NULL; q = q->next)
        {
            if (strcmp(q->ten_sach, min->ten_sach) < 0)
            {
                min = q;
            }
        }
        swapstring(min->ma_so_sach, p->ma_so_sach);
        swapstring(min->ten_sach, p->ten_sach);
    }
}
```

```
        swapstring(min->ten_tac_gia, p->ten_tac_gia);
        swapstring(min->nha_xuat_ban, p->nha_xuat_ban);
        swapint(&min->nam_xuat_ban, &p->nam_xuat_ban);
        swapint(&min->trang_thai_sach, &p->trang_thai_sach);
    }
    Xem_Toan_Bo_Sach(a);
}
```

```
void Theo_Van_Ten_Tac_Gia(book a)
{
    for (book p = a; p->next != NULL; p = p->next)
    {
        book min = p;
        for (book q = p->next; q != NULL; q = q->next)
        {
            if (strcmp(q->ten_tac_gia, min->ten_tac_gia) < 0)
            {
                min = q;
            }
        }
        swapstring(min->ma_so_sach, p->ma_so_sach);
        swapstring(min->ten_sach, p->ten_sach);
        swapstring(min->ten_tac_gia, p->ten_tac_gia);
        swapstring(min->nha_xuat_ban, p->nha_xuat_ban);
        swapint(&min->nam_xuat_ban, &p->nam_xuat_ban);
    }
}
```

```
        swapint(&min->trang_thai_sach, &p->trang_thai_sach);
    }
    Xem_Toan_Bo_Sach(a);
}

void Theo_Van_Nha_Xuat_Ban(book a)
{
    for (book p = a; p->next != NULL; p = p->next)
    {
        book min = p;
        for (book q = p->next; q != NULL; q = q->next)
        {
            if (strcmp(q->nha_xuat_ban, min->nha_xuat_ban) < 0)
            {
                min = q;
            }
        }
        swapstring(min->ma_so_sach, p->ma_so_sach);
        swapstring(min->ten_sach, p->ten_sach);
        swapstring(min->ten_tac_gia, p->ten_tac_gia);
        swapstring(min->nha_xuat_ban, p->nha_xuat_ban);
        swapint(&min->nam_xuat_ban, &p->nam_xuat_ban);
        swapint(&min->trang_thai_sach, &p->trang_thai_sach);
    }
    Xem_Toan_Bo_Sach(a);
}
```



```
}
```

```
void Theo_Ngay_Xuat_Ban(book a)
{
    for (book p = a; p->next != NULL; p = p->next)
    {
        book max = p;
        for (book q = p->next; q != NULL; q = q->next)
        {
            if (q->nam_xuat_ban > max->nam_xuat_ban)
            {
                max = q;
            }
        }
        swapstring(max->ma_so_sach, p->ma_so_sach);
        swapstring(max->ten_sach, p->ten_sach);
        swapstring(max->ten_tac_gia, p->ten_tac_gia);
        swapstring(max->nha_xuat_ban, p->nha_xuat_ban);
        swapint(&max->nam_xuat_ban, &p->nam_xuat_ban);
        swapint(&max->trang_thai_sach, &p->trang_thai_sach);
    }
    Xem_Toan_Bo_Sach(a);
}
```

```
void Xem_Danh_Sach_Theo_Thu_Tu(book a)
```

```
{
    int chose;
    do
    {
        printf("=====Xem danh sach theo thu tu=====");
        printf("\n1 .Theo van alphabet cua ten sach");
        printf("\n2 .Theo van alphabet cua ten tac gia");
        printf("\n3 .Theo van alphabet cua nha xuất ban");
        printf("\n4 .Sach duoc xuất ban moi nhat");
        printf("\n0 .Tro ve");
        printf("\n=====");
        printf("\n\nChose=");
        scanf("%d", &chose);
        switch (chose)
        {
            case 0:
                system("cls");
                break;
            case 1:
                system("cls");
                Theo_Van_Ten_Sach(a);
                break;
            case 2:
                system("cls");
                Theo_Van_Ten_Tac_Gia(a);
```

```
        break;
    case 3:
        system("cls");
        Theo_Van_Nha_Xuat_Ban(a);
        break;
    case 4:
        system("cls");
        Theo_Ngay_Xuat_Ban(a);
        break;
    default:
        printf("Sai chuc nang, vui long chon lai!\n");
        break;
    }
} while (chose);
}

void Xem_Sach(book a)
{
    int chose;
    do
    {
        printf("====Chuc nang xem danh sach====");
        printf("\n1 .Xem toan bo danh muc sach trong danh sach");
        printf("\n2 .Xem nhung cuon sach dang cho muon");
        printf("\n3 .Xem nhung cuon sach chua cho muon");
```

```
printf("\n4 .Xem danh sach theo thu tu");
printf("\n0 .Tro ve");
printf("\n=====");
printf("\n\nChose=");
scanf("%d", &chose);
switch (chose)
{
case 0:
    system("cls");
    break;
case 1:
    system("cls");
    Xem_Toan_Bo_Sach(a);
    break;
case 2:
    system("cls");
    Xem_Sach_Da_Muon(a);
    break;
case 3:
    system("cls");
    Xem_Sach_Chua_Muon(a);
    break;
case 4:
    system("cls");
    Xem_Danh_Sach_Theo_Thu_Tu(a);
```

```
        break;
    default:
        printf("Sai chuc nang, vui long chon lai!\n");
        break;
    }
} while (chose);
}
```

```
book Muon_sach(book a)
{
    if (a == NULL)
        return a;
    book p = a;
    infor s;
    Xem_Sach_Chua_Muon(a);
    printf("Nhap ma so sach muon muon:");
    fflush(stdin);
    gets(s);
    while (p != NULL)
    {
        if (strcmp(s, p->ma_so_sach) == 0 && p->trang_thai_sach == 1)
        {
            p->trang_thai_sach = 0;
            break;
        }
    }
}
```

```
        p = p->next;
    }
    return a;
}

book Tra_sach(book a)
{
    if (a == NULL)
        return a;
    book p = a;
    infor s;
    Xem_Sach_Da_Muon(a);
    printf("Nhap ma so sach muon tra:");
    fflush(stdin);
    gets(s);
    while (p != NULL)
    {
        if (strcmp(s, p->ma_so_sach) == 0 && p->trang_thai_sach == 0)
        {
            p->trang_thai_sach = 1;
            break;
        }
        p = p->next;
    }
    return a;
}
```

```
voidMuon_tra_sach(book a)
{
    int chose;
    do
    {
        system("cls");
        printf("====Chuc nang muon/tra sach====");
        printf("\n1 .Muon sach.");
        printf("\n2 .Tra sach.");
        printf("\n0 .Tro ve");
        printf("\n=====");
        printf("\n\nChose=");
        scanf("%d", &chose);
        switch (chose)
        {
            case 0:
                system("cls");
                break;
            case 1:
                system("cls");
                a = Muon_sach(a);
                break;
            case 2:
                system("cls");
```

```
        a = Tra_sach(a);
        break;
    default:
        printf("Sai chuc nang, vui long chon lai!\n");
        break;
    }
} while (chose);
}
```

```
book Sua_ten_sach(book a)
{
    infor s1, s2;
    printf("Nhap ma so sach:");
    fflush(stdin);
    gets(s1);
    printf("Nhap ten sach moi:");
    fflush(stdin);
    gets(s2);
    book p = a;
    int check = 0;
    while (p->next != NULL)
    {
        if (strcmp(p->ma_so_sach, s1) == 0)
        {
            check = 1;
        }
    }
}
```



```
        break;
    }
    p = p->next;
}
if (check == 1)
    strcpy(p->ten_sach, s2);
else
{
    printf("Ma so sach khong ton tai!\n");
}
return a;
}
```

```
book Sua_ten_tac_gia(book a)
{
    infor s1, s2;
    printf("Nhap ma so sach:");
    fflush(stdin);
    gets(s1);
    printf("Nhap ten tac gia moi:");
    fflush(stdin);
    gets(s2);
    book p = a;
    int check = 0;
    while (p->next != NULL)
```

```
{
    if (strcmp(p->ma_so_sach, s1) == 0)
    {
        check = 1;
        break;
    }
    p = p->next;
}
if (check == 1)
    strcpy(p->ten_tac_gia, s2);
else
{
    printf("Ma so sach khong ton tai!\n");
}
return a;
}
```

```
book Sua_nha_xuat_ban(book a)
{
    infor s1, s2;
    printf("Nhap ma so sach:");
    fflush(stdin);
    gets(s1);
    printf("Nhap nha xuất bản mới:");
    fflush(stdin);
```

```
    gets(s2);
    book p = a;
    int check = 0;
    while (p->next != NULL)
    {
        if (strcmp(p->ma_so_sach, s1) == 0)
        {
            check = 1;
            break;
        }
        p = p->next;
    }
    if (check == 1)
        strcpy(p->nha_xuat_ban, s2);
    else
    {
        printf("Ma so sach khong ton tai!\n");
    }
    return a;
}
```

```
book Sua_nam_san_xuat(book a)
{
    int year;
    infor s;
```

```
printf("Nhap ma so sach:");
fflush(stdin);
gets(s);
printf("Nhap nam san xuat moi:");
scanf("%d", &year);
book p = a;
int check = 0;
while (p->next != NULL)
{
    if (strcmp(p->ma_so_sach, s) == 0)
    {
        check = 1;
        break;
    }
    p = p->next;
}
if (check == 1)
    p->nam_xuat_ban = year;
else
{
    printf("Ma so sach khong ton tai!\n");
}
return a;
}
```

```
void Cap_nhat_noi_dung_cua_sach(book a)
{
    int chose;
    do
    {
        printf("====Chuc nang chinh sua noi dung cua
sach====");

        printf("\n1 .Sua ten sach.");
        printf("\n2 .Sua ten tac gia.");
        printf("\n3 .Sua nha xuất bản.");
        printf("\n4 .Sua năm xuất bản.");
        printf("\n0 .Tro về");

        printf("\n=====");

        printf("\n\nChose=");
        scanf("%d", &chose);
        switch (chose)
        {
            case 0:
                system("cls");
                break;
            case 1:
                system("cls");
                Xem_Toan_Bo_Sach(a);
                a = Sua_ten_sach(a);
                break;
```

case 2:

```
system("cls");  
Xem_Toan_Bo_Sach(a);  
a = Sua_ten_tac_gia(a);  
break;
```

case 3:

```
system("cls");  
Xem_Toan_Bo_Sach(a);  
a = Sua_nha_xuat_ban(a);  
break;
```

case 4:

```
system("cls");  
Xem_Toan_Bo_Sach(a);  
a = Sua_nam_san_xuat(a);  
break;
```

default:

```
printf("Sai chuc nang, vui long chon lai!\n");  
break;
```

```
}
```

```
} while (chose);
```

```
}
```

```
void save_data(book a)
```

```
{
```

```
infor s;
```

```
printf ("Nhap duong dan: ");
fflush(stdin);
gets(s);
FILE *f = fopen(s, "w+");
while (a != NULL)
{
    fputs(a->ma_so_sach, f);
    fputs(",", f);
    fputs(a->ten_sach, f);
    fputs(",", f);
    fputs(a->ten_tac_gia, f);
    fputs(",", f);
    fputs(a->nha_xuat_ban, f);
    fprintf(f, "\n%d %d\n", a->nam_xuat_ban, a->trang_thai_sach);
    a = a->next;
}
fclose(f);
}
```