

ĐẠI HỌC BÁCH KHOA
KHOA CÔNG NGHỆ THÔNG TIN



ĐỒ ÁN
GIẢI THUẬT VÀ LẬP TRÌNH

Đề tài 14:

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

GIẢNG VIÊN HƯỚNG DẪN: Võ Đức Hoàng

SINH VIÊN THỰC HIỆN: Hồ Ngọc Trung LỚP: 18T1 NHÓM: 18N10

Đà Nẵng, 12/2020

MỤC LỤC

LỜI MỞ ĐẦU	2
GIỚI THIỆU ĐỀ TÀI.....	3
1. CƠ SỞ LÝ THUYẾT	4
1.1. Ý tưởng	4
1.2. Cơ sở lý thuyết.....	5
2. TỔ CHỨC CẤU TRÚC DỮ LIỆU VÀ THUẬT TOÁN	5
2.1. Phát biểu bài toán	5
2.2. Cấu trúc dữ liệu	5
2.3. Thuật toán.....	6
3. CHƯƠNG TRÌNH VÀ KẾT QUẢ.....	8
3.1. Tổ chức chương trình.....	8
3.2. Ngôn ngữ cài đặt.....	9
3.3. Kết quả	11
3.3.1. Giao diện chính của chương trình	11
3.3.2. Kết quả thực thi của chương trình.....	13
3.3.3. Nhận xét.....	13
4. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN	13
4.1. Kết luận	13
4.2. Hướng phát triển.....	13
TÀI LIỆU THAM KHẢO.....	14

LỜI MỞ ĐẦU

Cấu trúc dữ liệu và Phân tích, thiết kế giải thuật là hai học phần rất quan trọng đối với người lập trình. Học phần này được xem như nền tảng của lập trình máy tính. Nó là cơ sở vững chắc để giải quyết nhiều bài toán, đồng thời cung cấp cho chúng ta hiểu biết về các giải thuật tác động lên dữ liệu, cũng như cách tổ chức dữ liệu hiệu quả để tối ưu bài toán.

Sau khi học xong học phần lý thuyết, em đã nghiên cứu và thực hiện đồ án này như là một cách để củng cố và mở rộng kiến thức. Thông qua quá trình thực hiện đồ án, chúng em đã nắm bắt được những kỹ thuật quan trọng của việc xây dựng cấu trúc dữ liệu và phân tích, thiết kế giải thuật sao cho tối ưu nhất.

Bài toán “Minh họa chương trình quản lý sách đơn giản trong thư viện” mà em nghiên cứu và trình bày trong báo cáo sau đây là một ví dụ.

Em xin chân thành cảm ơn thầy Võ Đức Hoàng đã bổ sung ý tưởng và tận tình giúp đỡ em thực hiện đồ án này. Em rất mong nhận được sự góp ý từ phía thầy để bài làm của em được hoàn thiện hơn.

GIỚI THIỆU 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.

A. Thông tin liên quan đến một cuốn sách gồm: Mã số sách, Tên sách, Tên tác giả, Nhà xuất bản, Năm xuất bản, Trạng thái sách: {TRUE: chưa mượn/ FALSE: đã mượn}

B. Các thao tác trên danh sách này:

1. Khởi tạo danh sách

- Khởi tạo danh mục sách rỗng (chưa có sách)
- Đọc từ file: nhập vào tên file đã lưu danh mục sách ở lần làm việc trước đó.

2. Thêm một cuốn sách vào danh sách

- Thêm vào đầu danh sách: InsertFirst
- Thêm vào sau một cuốn sách nào đó: InsertAfter
- Thêm vào cuối của danh sách: InsertLast

3. Xóa một cuốn sách khỏi danh sách theo

- Mã số
- Tên sách
- Tên tác giả: (nếu tác giả có nhiều sách thì xóa hết)
- Xóa cuốn sách ở đầu danh sách
- Xóa cuốn sách ở sau cuốn sách có mã số nào đó
- Xóa cuốn sách ở cuối danh sách

4. Tìm kiếm sách theo:

lưu ý kết quả có bao nhiêu thì liệt kê ra hết, không phải chỉ liệt kê kết quả đầu tiên

- Tên sách
- Tên tác giả
- Tên nhà xuất bản

5. Chức năng xem danh sách

- Xem toàn bộ danh mục sách trong danh sách
- Xem những cuốn sách đang cho mượn
- Xem những cuốn sách chưa cho mượn

d. Xem danh sách theo thứ tự:

- i.Theo vần alphabet của tên sách
- ii.Theo vần alphabet của tên tác giả
- iii.Theo vần alphabet của tên nhà xuất bản
- iv.Sách được xuất bản mới nhất (theo năm)

6. Chức năng mượn/trả sách

a. Mượn sách: liệt kê những cuốn sách chưa cho mượn, cho user chọn một cuốn sách \Rightarrow rồi cập nhật lại trạng thái cho mượn của sách.

b.Trả sách: nhập vào mã số sách được trả \Rightarrow cập nhật lại trạng thái đã trả sách cho cuốn sách đó.

7. Chức năng chỉnh sửa nội dung của sách: cho phép chọn các thông tin của sách để sửa và sau đó cập nhật lại.

8. Chức năng lưu file: nhập vào một tên file rồi lưu toàn bộ trạng thái hiện tại của danh mục sách vào đó.

Yêu cầu: Sinh viên phải thực hiện theo các yêu cầu sau:

- 1.Cài đặt cấu trúc dữ liệu Book theo mô tả như phần A
- 2. Cài đặt cấu trúc BookNode là phần tử của danh sách liên kết chứa danh mục sách
- 3. Cài đặt toàn bộ các chức năng mô tả trong phần B từ 1 -> 8: thể hiện các chức năng này theo menu chọn.

Ngoài ra sinh viên có thể bổ sung những chức năng mở rộng tùy ý. Tất cả các chức năng nâng cao này đều được đánh giá cao

1. CƠ SỞ LÝ THUYẾT

1.1. Ý tưởng

Với nhu cầu tri thức như hiện nay thì nhu cầu dùng Internet, báo chí hay đặc biệt là sách ngày càng không ngừng tăng cao. Khác với những nền tảng quản lý khác, quản lý sách cần quy hoạch một cách thật hợp lý để có thể dễ tìm kiếm mỗi khi cần và có thể kiểm soát mọi thứ một cách tốt nhất.

Để giải quyết vấn đề này thì việc đưa ra một chương trình giúp quản lý sách là một điều vô cùng quan trọng. Thiết kế một cấu trúc dữ liệu đánh giá được mức độ quan trọng cũng như mức độ ứng dụng của chương trình.

Việc ứng dụng kiến thức đã học vào thực tiễn là điều vô cùng quan trọng. Qua đó đề tài này được hình thành.

1.2. Cơ sở lý thuyết

- Cấu trúc dữ liệu: Mảng, Arraylist
- Giải thuật: Thuật toán sắp xếp, chèn, thuật toán thêm, sửa, xóa thông tin
- Kiến thức lập trình: ngôn ngữ lập trình Java
- Lập trình hướng đối tượng:
 - o Định nghĩa đối tượng mới với các kiểu dữ liệu được định nghĩa sẵn
 - o Khai thác điểm mạnh của lập trình hướng đối tượng như tính đóng gói, đa hình, kế thừa và tính trừu tượng.
 - o Xây dựng cấu trúc chương trình theo mô hình MVC (model-control-view) để dễ bảo trì, cập nhật cũng như chỉnh sửa code sau này

2. TỔ CHỨC CẤU TRÚC DỮ LIỆU VÀ THUẬT TOÁN

2.1. Phát biểu bài toán

- Dữ liệu vào (input):

Thông tin chung của sách bao gồm: Tên sách, Tên tác giả, Nhà xuất bản, Năm xuất bản, Trạng thái sách: {TRUE: chưa mượn/ FALSE: đã mượn}

- Dữ liệu ra (output):

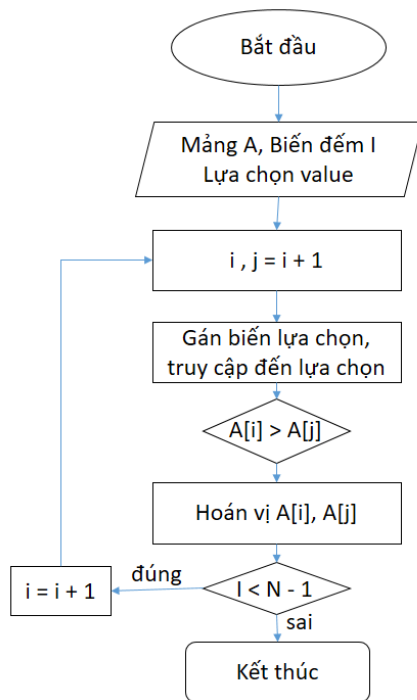
Menu thao tác cơ bản với sách như: thêm sách, xóa sách, cập nhật thông tin cho sách, mượn trả sách, tìm kiếm, chèn, hiển thị thông tin sách.

2.2. Cấu trúc dữ liệu

- Tất cả các đối tượng được tạo ra được lưu trong một mảng ArrayList với mỗi phần tử trong mảng là một đối tượng sách.
- Khi gọi đến đối tượng thì ta trực tiếp thao tác trên mảng dữ liệu
- Các hàm sẽ sử dụng trực tiếp dữ liệu có trong database, các thay đổi đều sẽ được lưu trong lại.
- Dữ liệu trong file sẽ không bị ảnh hưởng trừ khi ta lưu lại dữ liệu. Dữ liệu sẽ được ghi đè lên dữ liệu cũ, tránh trùng lặp với dữ liệu cũ.

2.3. Thuật toán

- Thuật toán sắp xếp (sort):



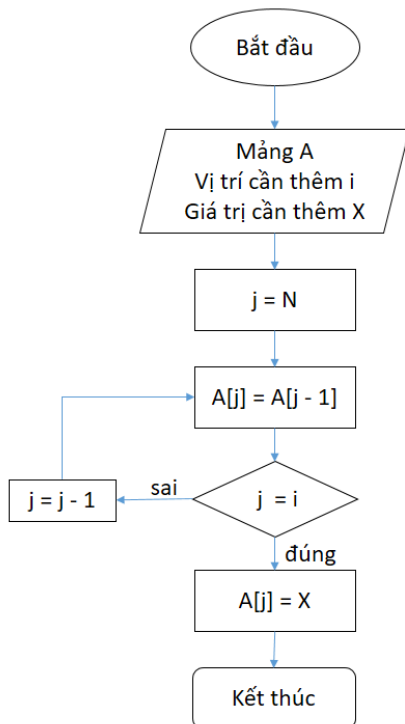
- Thuật toán bắt đầu từ khi ta truyền mảng dữ liệu vào.

- Dữ liệu sẽ được truyền đến lựa chọn ta đã đặt trong tham số truyền vào.

- Nếu thỏa thì sẽ thực hiện việc hoán đổi vị trí, nếu không thì sẽ không thực hiện mà bỏ qua để đến lần lặp tiếp theo.

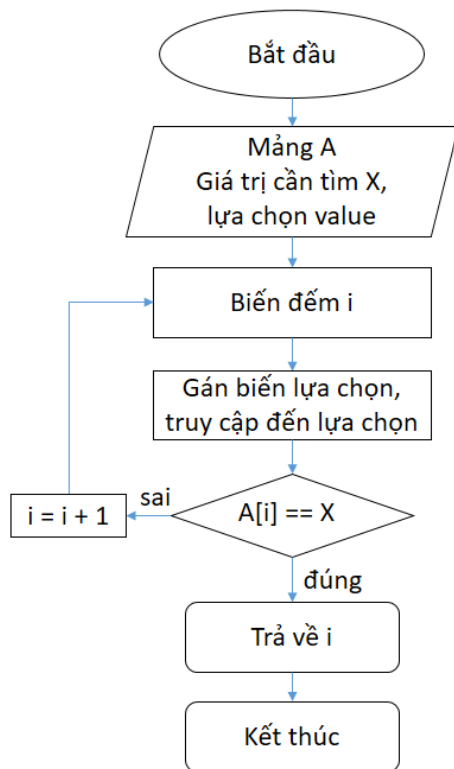
- Thuật toán trả về Mảng đã được sắp xếp

- Thuật toán thêm (insert):



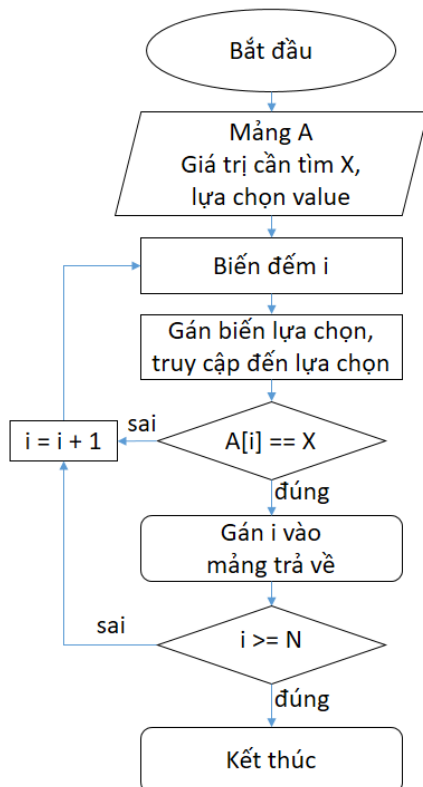
- Thuật toán trả về Mảng đã được cập nhật thêm phần tử vào vị trí

- Thuật toán tìm kiếm 1 phần tử (search):



- Thuật toán trả về vị trí cần tìm

- Thuật toán tìm kiếm toàn bộ phần tử (searchAll):

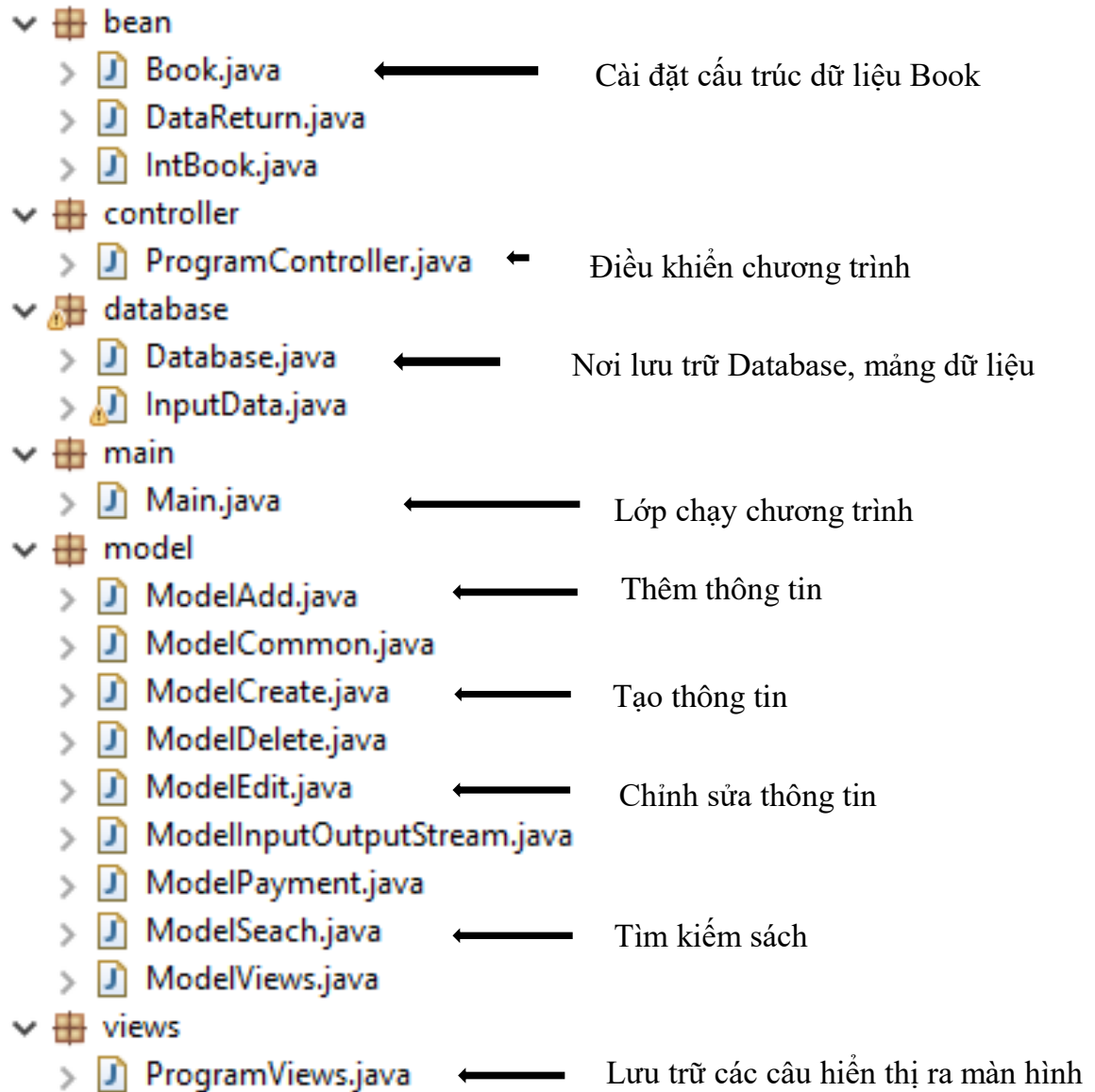


- Thuật toán trả về mảng vị trí thỏa điều kiện cần tìm

3. CHƯƠNG TRÌNH VÀ KẾT QUẢ

3.1. Tổ chức chương trình

Gồm các class được chia theo mô hình MVC cho dễ quản lý cũng như bảo trì code, gồm các package:



Các lớp sẽ được gọi từ lớp `Main.java` theo trình tự:

- Lớp `InputData.java` sẽ nhận thông tin từ bàn phím
- Tiếp theo là lớp `ProgramController.java` sẽ nhận thông tin từ lớp `InputData.java`, sẽ gọi đến các lớp trong package `model`
- `Model` sẽ gọi `views` để hiển thị thông tin cần thiết sau đó trả lại cho lớp `Main.java`

3.2. Ngôn ngữ cài đặt

- *Hàm tìm kiếm thông tin:* Truyền vào mảng Loại thông tin Giá trị

Lựa chọn loại thông tin cần tìm kiếm

```
// Ham tìm kiếm thông tin
public int search(ArrayList<bean.Book> book, String type, String value) {
    int indexReturn = -1;
    for (int i = 0; i < getSize(book); i++) {
        switch (type) {
            case "id":
                book.get(i).setValue(book.get(i).getId_Book());
                break;
            case "name":
                book.get(i).setValue(book.get(i).getName_Book());
                break;
            case "author":
                book.get(i).setValue(book.get(i).getName_Author());
                break;
            case "publisher":
                book.get(i).setValue(book.get(i).getName_Publish());
                break;
            case "year":
                book.get(i).setValue(book.get(i).getYear_Publishing());
                break;
            default:
                break;
        }
        if (book.get(i).getValue().equals(value)) {
            indexReturn = i;
            break;
        }
    }
    return indexReturn;
}
```

Trùng thì lưu lại vị

Trả lại vị trí

- *Hàm hoán đổi vị trí:*

```
// Ham hoan doi hai phan tu trong danh sach
public void swap(ArrayList<bean.Book> book, int i, int j) {
    bean.Book temp = book.get(i);
    book.set(i, book.get(j));
    book.set(j, temp);
}
```

- Hàm sắp xếp thông tin:

Truyền vào Mảng loại cần sắp xếp

```
// Ham sap xep danh sach
public ArrayList<bean.Book> sort(ArrayList<bean.Book> book, String type) {
    for (int i = 0; i < getSize(book) - 1; i++) {
        for (int j = i + 1; j < getSize(book); j++) {
            switch (type) {
                case "name":
                    if (book.get(i).getName_Book().compareTo(book.get(j).getName_Book()) > 0) {
                        swap(book, i, j);
                    }
                    break;
                case "author":
                    if (book.get(i).getName_Author().compareTo(book.get(j).getName_Author()) > 0) {
                        swap(book, i, j);
                    }
                    break;
                case "publisher":
                    if (book.get(i).getName_Publish().compareTo(book.get(j).getName_Publish()) > 0) {
                        swap(book, i, j);
                    }
                    break;
                case "year":
                    if (book.get(i).getYear_Publishing().compareTo(book.get(j).getYear_Publishing()) < 0) {
                        swap(book, i, j);
                    }
                    break;
            }
        }
    }
    return book;
}
```

Chọn loại cần sắp xếp

Hoán đổi vị trí

Nếu lớn hơn

Trả lại mảng đã sắp xếp

- Hàm chèn thông tin:

Mảng dữ liệu dữ liệu cần chèn vị trí

```
// Ham chen vao tung vi tri
public void addBook(ArrayList<bean.Book> book, bean.Book addBook, int index) {
    int size = getSize(book);
    or (int i = index; i < size; i++) {
        if (i == index) {
            book.add(addBook);
        }
        book.add(book.get(i));
    }
    for (int j = index; j < size; j++) {
        book.remove(index);
    }
}
```

Độ lớn của mảng

Thêm phần tử vào vị trí

Thêm các phần tử phía sau vào mảng

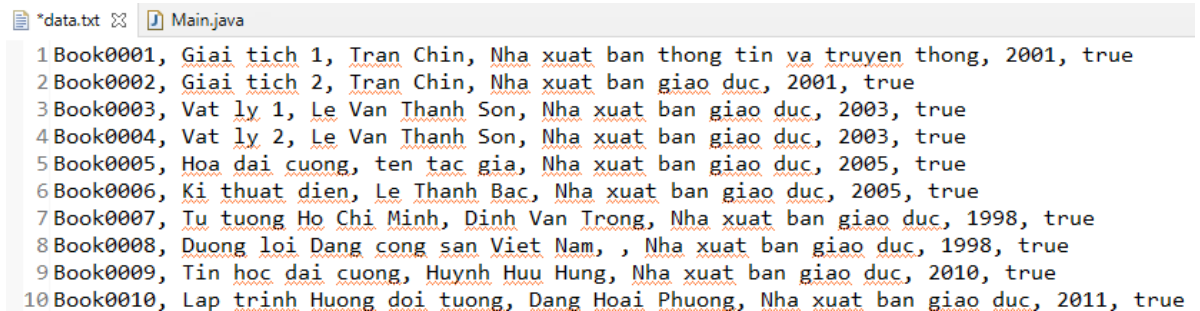
Xóa các phần tử từ vị trí vừa thêm vào

3.3. Kết quả

3.3.1. Giao diện chính của chương trình

```
-----Menu-----
1. Khoi tao
-----end!-----
1
Doc file da luu
Nhap ten file :
data.txt
Thanh cong
-----Menu-----
2. Them mot cuon sach vao danh sach
3. Xoa mot cuon sach khoi danh sach
4. Tim kiem danh sach
5. Xem danh sach
6. Muon tra sach
7. Chinh sua noi dung sach
8. Luu file
-----end!-----
```

- Ví dụ ta có một file dữ liệu như sau:



```
*data.txt  Main.java
1Book0001, Giai tich 1, Tran Chin, Nha xuất bản thông tin và truyền thông, 2001, true
2Book0002, Giai tich 2, Tran Chin, Nha xuất bản giao duc, 2001, true
3Book0003, Vat ly 1, Le Van Thanh Son, Nha xuất bản giao duc, 2003, true
4Book0004, Vat ly 2, Le Van Thanh Son, Nha xuất bản giao duc, 2003, true
5Book0005, Hoa dai cuong, ten tac gia, Nha xuất bản giao duc, 2005, true
6Book0006, Ki thuat dien, Le Thanh Bac, Nha xuất bản giao duc, 2005, true
7Book0007, Tu tuong Ho Chi Minh, Dinh Van Trong, Nha xuất bản giao duc, 1998, true
8Book0008, Duong loi Dang cong san Viet Nam, , Nha xuất bản giao duc, 1998, true
9Book0009, Tin hoc dai cuong, Huynh Huu Hung, Nha xuất bản giao duc, 2010, true
10Book0010, Lap trinh Huong doi tuong, Dang Hoai Phuong, Nha xuất bản giao duc, 2011, true
```

- Khi ta chạy chương trình, ta chọn mục xem danh sách, chọn xem toàn bộ danh mục theo thứ tự tên sách ta được như sau:

```
-----Menu-----
1. Khoi tao
-----end!-----
1
Doc file da luu
Nhap ten file :
data.txt
Thanh cong
-----Menu-----
2. Them mot cuon sach vao danh sach
3. Xoa mot cuon sach khoi danh sach
4. Tim kiem danh sach
5. Xem danh sach
6. Muon tra sach
7. Chinh sua noi dung sach
8. Luu file
-----end!-----
5
1. Xem toan bo danh muc trong sach
2. Xem nhung cuong sach dang cho muon
3. Xem nhung cuong sach chua cho muon
4. Xem danh sach theo thu tu
4
1. Xem theo van alphabet theo ten sach
2. Xem theo van alphabet theo ten tac gia
3. Xem theo van alphabet theo ten nha xuất bản
4. Sach duoc xuất bản moi nhât
1
Ma so sach: Book0008
Ten sach: Duong loi Dang cong san Viet Nam
Ten tac gia:
len tac gia:
Nha xuất bản: Nha xuất bản giao duc
Nam xuất bản: 1998
Trang thai: false
-----
Ma so sach: Book0001
Ten sach: Giai tich 1
Ten tac gia: Tran Chin
Nha xuất bản: Nha xuất bản thông tin và truyền thông
Nam xuất bản: 2001
Trang thai: false
-----
Ma so sach: Book0002
Ten sach: Giai tich 2
Ten tac gia: Tran Chin
Nha xuất bản: Nha xuất bản giao duc
Nam xuất bản: 2001
Trang thai: false
-----
Ma so sach: Book0005
Ten sach: Hoa dai cuong
Ten tac gia: ten tac gia
Nha xuất bản: Nha xuất bản giao duc
Nam xuất bản: 2005
Trang thai: false
-----
Ma so sach: Book0006
Ten sach: Ki thuat dien
Ten tac gia: Le Thanh Bac
Nha xuất bản: Nha xuất bản giao duc
Nam xuất bản: 2005
Trang thai: false
```

- Hoặc khi ta chọn vào tìm kiếm sách theo tên sách:

Nếu ta nhập sai

```
-----Menu-----
2. Them mot cuon sach vao danh sach
3. Xoa mot cuon sach khoi danh sach
4. Tim kiem danh sach
5. Xem danh sach
6. Muon tra sach
7. Chinh sua noi dung sach
8. Luu file
-----end!-----
```

```
4
1. Tim kiem theo ten sach
2. Tim kiem theo ten tac gia
3. Tim kiem theo ten nha xuất ban
```

```
1
Nhap ten cuon sach :
ten sach không dung
```

```
|-----Menu-----
2. Them mot cuon sach vao danh sach
3. Xoa mot cuon sach khoi danh sach
4. Tim kiem danh sach
5. Xem danh sach
6. Muon tra sach
7. Chinh sua noi dung sach
8. Luu file
-----end!-----
```

That bai!

Nếu ta nhập đúng

```
-----Menu-----
2. Them mot cuon sach vao danh sach
3. Xoa mot cuon sach khoi danh sach
4. Tim kiem danh sach
5. Xem danh sach
6. Muon tra sach
7. Chinh sua noi dung sach
8. Luu file
-----end!-----
```

```
4
1. Tim kiem theo ten sach
2. Tim kiem theo ten tac gia
3. Tim kiem theo ten nha xuất ban
```

```
1
Nhap ten cuon sach :
Vat ly 1
Ma so sach: Book0003
Ten sach: Vat ly 1
Ten tac gia: Le Van Thanh Son
Nha xuất ban: Nha xuất ban giao duc
Nam xuất ban: 2003
Trang thai: false
-----end!-----
```

Thanh cong

- Khi ta lựa chọn thêm thông tin vào một vị trí nhất định, cụ thể là sau cuốn sách tên Giai tích 1, ta được như sau:

Nếu ta nhập sai tên sách

```
-----Menu-----
2. Them mot cuon sach vao danh sach
3. Xoa mot cuon sach khoi danh sach
4. Tim kiem danh sach
5. Xem danh sach
6. Muon tra sach
7. Chinh sua noi dung sach
8. Luu file
-----end!-----
2
1. Them vao dau danh sach
2. Them vao sau cuon sach
3. Them vao cuoi danh sach
2
Nhap ten cuon sach :
trung
Ten vua nhap không trung voi cuon sach nao!
That bai!
```

```
-----Menu-----
2. Them mot cuon sach vao danh sach
3. Xoa mot cuon sach khoi danh sach
4. Tim kiem danh sach
5. Xem danh sach
6. Muon tra sach
7. Chinh sua noi dung sach
8. Luu file
-----end!-----
```

Nếu ta nhập đúng tên sách

```
6. Muon tra sach
7. Chinh sua noi dung sach
8. Luu file
-----end!-----
```

```
2
1. Them vao dau danh sach
2. Them vao sau cuon sach
3. Them vao cuoi danh sach
```

```
2
Nhap ten cuon sach :
Giai tích 1
Nhap ma so:
Bookthem
Nhap ten sach:
Do an giai thuat lap trinh
Nhap ten tac gia:
Vo Duc Hoang
Nhap ten nha xuất ban:
DHBK
Nhap nam xuất ban:
2020
Thanh cong
```

```
-----Menu-----
```

3.3.2. Kết quả thực thi của chương trình

Kết quả của chương trình là chính xác, trong quá trình debug kiểm tra lỗi thì chưa gặp lỗi vặt nào.

3.3.3. Nhận xét

- Thuật toán sử dụng vẫn chưa được tối ưu đối với dữ liệu lớn.
- Chương trình còn khá chậm khi lượng dữ liệu đầu vào quá lớn.

4. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

4.1. Kết luận

- Chương trình tìm được kết quả một cách chính xác khi dữ liệu đầu vào không quá lớn.
- Biết cách nghĩ ra thuật toán và triển khai thuật toán.
- Biết được cách viết dự án theo mô hình MVC để dễ kiểm tra lỗi cũng như bảo trì code sau này.

4.2. Hướng phát triển

- Triển khai với thuật toán khác tối ưu hơn, tốc độ cao hơn.
- Tạo thêm giao diện GUI thân thiện với người dùng.
- Tối ưu code để code được tận dụng cũng như code gọn gàng sạch sẽ hơn.

TÀI LIỆU THAM KHẢO

- Giáo trình phân tích thiết kế và giải thuật
- Giáo trình cấu trúc dữ liệu
- Giáo trình java
- Nguồn Internet