**TRƯỜNG ĐẠI HỌC TÀI NGUYÊN VÀ MÔI TRƯỜNG THÀNH PHỐ HỒ CHÍ MINH**



**KHOA HỆ THỐNG THÔNG TIN VÀ VIỄN THÁM**



**BÁO CÁO ĐỒ ÁN MÔN LẬP TRÌNH HĐT**

**Quản Lý Kho Sách Giáo Khoa**

|  |  |
| --- | --- |
| Hồ Tuấn Khanh | 1050080103 |
| Nguyễn Thị Thanh Nhi | 1050080110 |
| Nguyễn Sơn Lâm | 1050080105 |

Lớp: K10\_ĐH\_CNTT3

***TP. Hồ Chí Minh, tháng 04 năm 2023***

**TRƯỜNG ĐẠI HỌC TÀI NGUYÊN VÀ MÔI TRƯỜNG THÀNH PHỐ HỒ CHÍ MINH**

**KHOA HỆ THỐNG THÔNG TIN VÀ VIỄN THÁM**



**BÁO CÁO ĐỒ ÁN MÔN LẬP TRÌNH HĐT**

**Quản Lý Kho Sách Giáo Khoa**

|  |  |
| --- | --- |
| Hồ Tuấn Khanh | 1050080103 |
| Nguyễn Thị Thanh Nhi | 1050080110 |
| Nguyễn Sơn Lâm | 1050080105 |

Lớp: K10\_ĐH\_CNTT3

***TP. Hồ Chí Minh, tháng 04 năm 2023***

**Lời nói đầu**

Đồ án lập trình phần mềm quản lý kho sách giáo khoa là một dự án được thực hiện trong môn học Lập trình hướng đối tượng. Đây là một đề tài có tính thực tiễn cao, đáp ứng nhu cầu quản lý và sử dụng sách giáo khoa của các trường học.

Đồ án bao gồm việc thiết kế và xây dựng một ứng dụng quản lý kho sách giáo khoa bằng ngôn ngữ lập trình Java, với các chức năng cơ bản như nhập, xuất sách, tìm kiếm sách theo tên, mã sách, thể loại, cập nhật thông tin sách, quản lý thông tin nhân viên, quản lý thông tin khách hàng và thống kê doanh thu.

Qua đồ án này, sinh viên có cơ hội rèn luyện kỹ năng lập trình hướng đối tượng, thiết kế và triển khai ứng dụng thực tế. Ngoài ra, đồ án cũng giúp sinh viên hiểu rõ hơn về quy trình phát triển phần mềm và áp dụng các kiến thức đã học trong môn học vào thực tế.

**Mục Lục**

**[I. Phát biểu bài toán 1](#_Toc31713)**

**[2. Class SachGiaoKhoa: 2](#_Toc9532)**

**[3. Class SachGiaoKhoa\_Cap1 kế thừa từ class SachGiaoKhoa: 2](#_Toc24313)**

**[4. Class SachGiaoKhoa\_Cap2 kế thừa từ class SachGiaoKhoa: 3](#_Toc4486)**

**[5. Class SachGiaoKhoa\_Cap3 kế thừa từ class SachGiaoKhoa: 3](#_Toc5889)**

**[6. Class QuanLyKhoSach 4](#_Toc14807)**

**[7. Class Menu 5](#_Toc22385)**

**[8. Class Test 5](#_Toc19587)**

**[9. Mô hình UML 6](#_Toc8986)**

**[III.Mô Tả Thao Tác Thuật Toán 6](#_Toc24479)**

**[IV.Cài đặt bài toán 7](#_Toc1059)**

**[2. Class SachGiaoKhoa 10](#_Toc30607)**

**[3. Class SachGiaoKhoa\_Cap1 13](#_Toc27855)**

**[4. Class SachGiaoKhoa\_Cap2 19](#_Toc1846)**

**[5. Class SachGiaoKhoa\_Cap3 24](#_Toc20343)**

**[6. Class QuanLyKhoSach 30](#_Toc24195)**

**[7. Class Menu 38](#_Toc12133)**

**[8. Class Test 43](#_Toc25187)**

**[V.Kiểm thử 44](#_Toc14701)**

**[1. Nhập thông tin 45](#_Toc365)**

**[2. Xuất thông tin của kho tổng 47](#_Toc10881)**

**[3.Xuất thông tin của kho cấp 1 47](#_Toc12992)**

**[3. Xuất thông tin của kho cấp 2 47](#_Toc15236)**

**[5.xuất thông tin kho cấp 3 47](#_Toc30947)**

**[6. Kiểm tra kho có rỗng không 48](#_Toc17714)**

**[7.xuất ra số lượng tổng 48](#_Toc28850)**

**[8.xuất ra số lượng từng loại 48](#_Toc9178)**

I. Phát biểu bài toán  
 Thư viện nhà trường là nơi chứa nhiều loại sách dành cho học sinh và sinh viên, có hàng trăm, ngàn nghìn loại sách và tên sách khác nhau, nhiều loại sách dành cho nhiều đối tượng học sinh cấp 1, cấp 2, cấp 3. Vậy làm thế nào để giải bài toán quản lý sách giáo khoa. Đây là đề tài của nhóm chúng em, giải đáp trong những phần dưới đây. Sách giáo khoa bao gồm: SGK cấp 1, SGK cấp 2, SGK cấp 3. Các đối tượng cần quản lý:

Sách Giáo Khoa: là lớp trừu tượng mỗi sách giáo khoa gồm có mã sách, nhà xuất bản, ngày xuất bản, ngày nhập kho, số lần tái bản, số lượng.

Sách Giáo Khoa Cấp 1, Cấp 2, Cấp 3: một sách giáo khoa có một tên sách riêng. Mỗi sách được thừa kế đối tượng Sách Giáo Khoa bao gồm mã sách, nhà xuất bản, ngày xuất bản, ngày nhập kho, số lần tái bản, số lượng.

ArrayList<SachGiaoKhoa> quản lý cả lớp SGK cấp 1, cấp 2 ,cấp 3 và Người, khi nhập ta có thể nhập tùy theo đối tượng SGK cấp1 hoặc SGK cấp 2 hoặc SGK cấp 3 vào ArrayList này

Tính kế thừa: SGK cấp1, cấp2, SGK cấp 3 cùng có các thuộc tính như mã sách, nhà xuất bản, ngày xuất bản, ngày nhập kho, số lần tái bản, số lượng.

Tính đa hình: SGK cấp1, cấp2, cấp 3 đều là thể hiện đối tượng SGK.

## II.Xác định các lớp của bài toán 1. **Class Date :**

* Thuộc tính : int ngay, int thang ,int nam
* Phương thức :
* nhapDate() : dùng để nhập ngày tháng năm có ràng buộc ( ví dụ tháng 1 chỉ có 31 ngày )

## Class SachGiaoKhoa:

* Thuộc tính: String maSach,String tenSach, String nhaXuatBan, Date ngayXuatBan, Date ngayNhapKho, int soLanTaiBan, int soLuong
* Phương thức:
* Constructor: Hàm tạo có tham số và hàm tạo không tham số
* Getter, Setter: Phương thức get, set cho các thuộc tính kiểu protected
* nhap(): nhập thông tin các thuộc tính
* chinhSuaThongTin() : dùng để chỉnh sửa thông tin thuộc tính trừ mã sách
* toString: Xuất ra các thông tin
* Phương thức trừu tượng : kiểm tra sách mới

## Class SachGiaoKhoa\_Cap1 kế thừa từ class SachGiaoKhoa:

* Phương thức:
* nhap(): override nhap() ở class SachGiaoKhoa và nhập thêm thuộc tính riêng của lớp
* chinhSuaThongTin() override của class SachGiaoKhoa và chỉnh sửa thêm thuộc tính của lớp này
* Getter, Setter: Phương thức get, set cho các thuộc tính kiểu private
* Kiểm tra sách mới : ghi đè của phương thức trừu tượng của lớp cha (nếu sách có năm lớn hơn 2021 thì là sách mới )
* toString(): override toString() ở class SachGiaoKhoa, hiện ra các thông tin của lớp SGK và SGK cấp 1.

## Class SachGiaoKhoa\_Cap2 kế thừa từ class SachGiaoKhoa:

* Phương thức:
* nhap(): override nhap() ở class SachGiaoKhoa
* chinhSuaThongTin() override của class SachGiaoKhoa và chỉnh sửa thêm thuộc tính của lớp này
* Getter, Setter: Phương thức get, set cho các thuộc tính kiểu private
* Kiểm tra sách mới : ghi đè của phương thức trừu tượng của lớp cha (nếu sách có năm lớn hơn 2021 thì là sách mới )
* toString(): override toString() ở class SachGiaoKhoa, hiện ra các thông tin của lớp SGK và SGK cấp 2.

## Class SachGiaoKhoa\_Cap3 kế thừa từ class SachGiaoKhoa:

* Phương thức:
* nhap(): override nhap() ở class SachGiaoKhoa
* chinhSuaThongTin() override của class SachGiaoKhoa và chỉnh sửa thêm thuộc tính của lớp này
* Getter, Setter: Phương thức get, set cho các thuộc tính kiểu private
* Kiểm tra sách mới : ghi đè của phương thức trừu tượng của lớp cha (nếu sách có năm lớn hơn 2021 thì là sách mới )
* toString(): override toString() ở class SachGiaoKhoa, hiện ra các thông tin của lớp SGK và SGK cấp 3.

## Class QuanLyKhoSach

* Thuộc tính :ArrayList<SachGiaoKhoa>list
* Phương thức :
  + Constructor : có tham số và không tham số
  + themVaoDanhSach(SachGiaoKhoa sgk) :thêm đối tượng( được truyền vào ) có thể là sách khoa cấp 1 hoặc cấp 2 hoặc cấp 3, nhưng không thể thêm vào một đối tượng trùng mã sách vào danh sách
  + inDanhSach() :in tất cả đối tượng trong danh sách
  + kiemTraDanhSachRong(): kiểm tra danh sách có rỗng hay không
  + soLuongSachHienCo() : in ra số lượng sách giáo khoa hiện có
  + soLuongSachGiaoKhoa\_Cap1() : in ra số lượng sách giáo khóa cấp 1 trong danh sách
  + soLuongSachGiaoKhoa\_Cap2() : in ra số lượng sách giáo khóa cấp 2 trong danh sách
  + soLuongSachGiaoKhoa\_Cap3() : in ra số lượng sách giáo khóa cấp 3 trong danh sách
  + lamRongDanhSach() : làm rỗng danh sách hiện có
  + kiemTraTonTai\_TheoMaSach(String maSach) : nhập mã sách vào và kiểm tra xem sách đó có tồn tại hay không
  + timSach\_TheoTen(String tenSach): nhập tên sách cần tìm và in ra thông tin nếu trùng tên
  + xoaSach\_TheoTen(String maSach): nhập mã sách và nếu tìm thấy mã sách trùng với mã sách đó thì xóa
  + sapXepTangDan\_TheoSoLuong(): sắp xếp tăng dần theo số lượng (sử dụng collections , comparator )
  + sapXepGiamDan\_TheoSoLuong(): sắp xếp giảm dần theo số lượng (sử dụng collections , comparator )
  + sapXepTangDan\_TheoMaSach(): sắp xếp tăng dần theo mã sách (sử dụng collections và sử dụng compareTo ở lớp mỗi lớp con)
  + sapXepGiamDan\_TheoMaSach(): sắp xếp giảm dần theo mã sách (sử dụng collections , comparator )
  + inSachCap1() : in ra tất cả sách giáo khoa cấp 1
  + inSachCap2() : in ra tất cả sách giáo khoa cấp 2
  + inSachCap3() : in ra tất cả sách giáo khoa cấp 3
  + xoaSach\_TheoTenSach() : nhập tên sách và xóa sách nào có tên sách trùng với tên sách đã nhập
  + chỉnhSuaThongTin\_TheoMa() : nhập mã sách vào nếu có mã sách trùng với mã sách đã nhập thì được phép chỉnh sửa các thông tin còn lại trừ mã sách

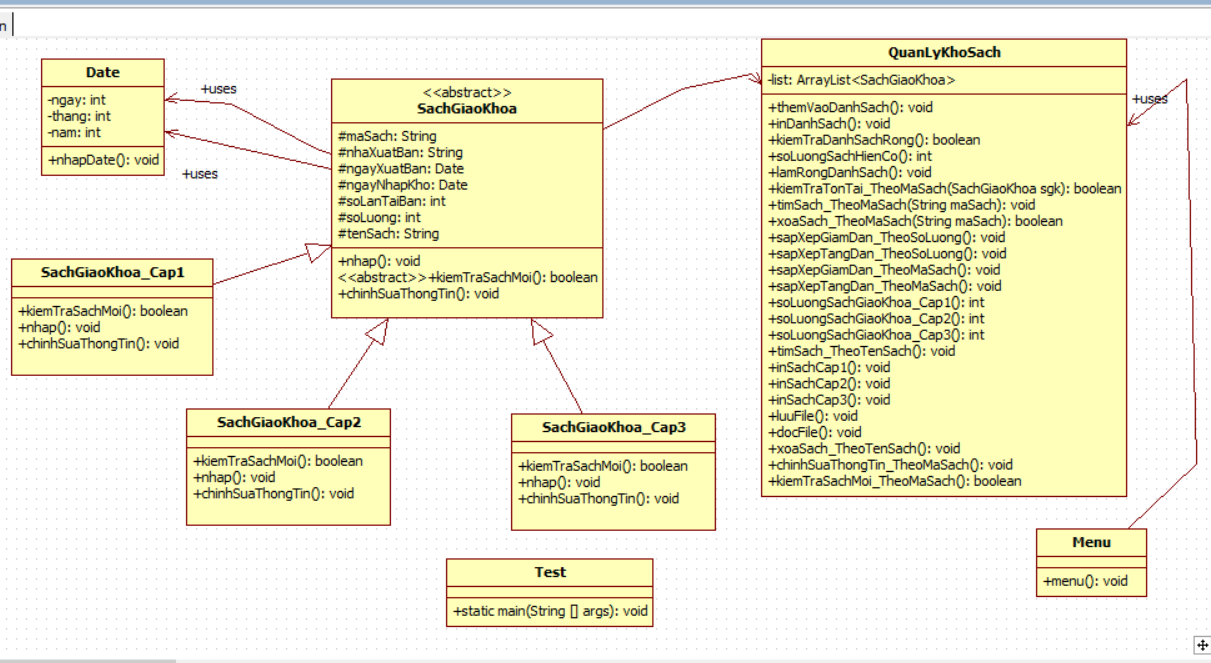
## Class Menu

* Phương Thức : Hiện ra menu để Chọn

## Class Test

* Phương Thức : dùng hàm Main để khởi động chương trình và gọi class Menu vào thực hiện chương trình.

## Mô hình UML

* 

**III.Mô Tả Thao Tác Thuật Toán**  
1. **Chỉnh sửa thông tin sách giáo khoa cấp 1 ,cấp 2 ,cấp 3**Nhập mã sách cần tìm để chỉnh sửa :

Dùng vòng lặp để so sánh mã sánh nếu trùng thì dùng hàm chỉnh sửa thông tin đã định nghĩa ở lớp cha và lớp con ( không được chỉnh sửa mã sách )

Thêm : nhập thông tin đối tượng sách giáo khoa cấp 1,2,3 và thêm vào danh sách ( đã ràng buộc khi tìm thấy trùng mã đã nhập sẵn trong danh sách thì không cho nhập )

Xóa : dùng phép remove trong ArrayList (xóa theo tên , theo mã sách )

## Hiện ra danh sách sách , sách cấp 1 ,2 ,3, hiện theo tên, mã sách

Thuật toán :

Trong ArrayList đã nhập tìm kiếm và hiện ra những thi sinh theo mã sách , theo tên đã nhập và xuất ra danh sách theo sách giáo khoa cấp 1,2,3 , tất cả các sách

## Sắp xếp giảm dần ,tăng dần theo mã sách , theo số lượng của sách

Sử dụng compareTo đã được định nghĩa trong các lớp con và dùng Collections.sort để sắp xếp theo mã sách tăng dần

Sử dụng Collections.sort+comparator để định nghĩa lại cách sắp xếp giảm dần của mã sách,số lượng sách và tăng dần của số lượng sách

Ví dụ cần sắp xếp tăng dần theo số lượng thì số lượng nhỏ phải đứng trước thì đối tượng nào có số lượng nhỏ thì return -1

## 4. Kiểm tra danh sách ( tồn tại , rỗng , làm rỗng danh sách )

5. Đọc File , Ghi File

Sử dụng ObjectInputStream,ObjectOutputStream,FileInPutStream,FileOutputStream

Truyền Đường dẫn file chỗ nào mình muốn ghi file và đọc file

# IV.Cài đặt bài toán

## 1. Class date

package sach;

import java.util.Scanner;

public class Date {

private int ngay, thang, nam;

public Date(int ngay, int thang, int nam) {

this.ngay = ngay;

this.thang = thang;

this.nam = nam;

}

public Date() {

}

public int getNgay() {

return ngay;

}

public void setNgay(int ngay) {

this.ngay = ngay;

}

public int getThang() {

return thang;

}

public void setThang(int thang) {

this.thang = thang;

}

public int getNam() {

return nam;

}

public void setNam(int nam) {

this.nam = nam;

}

public void nhapDate() {

// Ràng buộc nếu năm <=0 thì nhập lại

Scanner sc = new Scanner(System.in);

do {

System.out.println("Nhập năm của sách");

this.setNam(sc.nextInt());

if (this.getNam() <= 0 && this.getNam() > 2023) {

System.out.println("Vui lòng nhập năm của sách lớn hơn 0 !");

}

} while (this.getNam() <= 0);

// Ràng buộc nếu tháng <=0 hoặc >12 thì nhập lại

do {

System.out.println("Nhập tháng của sách");

this.setThang(sc.nextInt());

if (this.getThang() <= 0 || this.getThang() > 12) {

System.out.println("Vui lòng nhập tháng của sách lớn hơn 0 và nhỏ hơn 13 !");

}

} while (this.getThang() <= 0 || this.getThang() > 12);

// Ràng buộc ngày

// Tháng 1,3,5,7,8,10,12 => 0 < ngày < 32

// Tháng 2 năm nhuận => 0 < ngay < 30

// Tháng 2 năm không nhuận => 0 < ngay <29

// Tháng còn lại là => 0 < ngay < 31

if (this.getThang() == 1 || this.getThang() == 3 || this.getThang() == 5 || this.getThang() == 7

|| this.getThang() == 8 || this.getThang() == 10 || this.getThang() == 12) {

do {

System.out.println("Nhập ngày ");

this.setNgay(sc.nextInt());

if (this.getNgay() <= 0 || this.getNgay() > 31) {

System.out.println("Vui lòng nhập ngày từ 1-31 !");

}

} while (this.getNgay() <= 0 || this.getNgay() > 31);

} else if (this.getThang() == 2) {

if (this.getNam() % 400 == 0 || this.getNam() % 4 == 0 && this.getNam() % 100 != 0) {

do {

System.out.println("Nhập ngày ");

this.setNgay(sc.nextInt());

if (this.getNgay() <= 0 || this.getNgay() > 29)

System.out.println("Vui Lòng nhập ngày từ 1-29 !");

} while (this.getNgay() <= 0 || this.getNgay() > 29);

} else {

do {

System.out.println("Nhập ngày ");

this.setNgay(sc.nextInt());

if (this.getNgay() <= 0 || this.getNgay() > 28)

System.out.println("Vui Lòng nhập ngày từ 1-28 !");

} while (this.getNgay() <= 0 || this.getNgay() > 28);

}

} else {

do {

System.out.println("Nhập ngày");

this.setNgay(sc.nextInt());

if (this.getNgay() <= 0 || this.getNgay() > 30)

System.out.println("Vui lòng nhập ngày từ 1-30 !");

} while (this.getNgay() <= 0 || this.getNgay() > 30);

}

}

@Override

public String toString() {

return this.getNgay() + "/" + this.getThang() + "/" + this.getNam();

}

}

## Class SachGiaoKhoa

package sach;

import java.io.Serializable;

import java.util.Objects;

import java.util.Scanner;

public abstract class SachGiaoKhoa implements Comparable<SachGiaoKhoa>,Serializable {

protected String maSach,tenSach, nhaXuatBan;

protected Date ngayXuatBan, ngayNhapKho;

protected int soLanTaiBan, soLuong;

public SachGiaoKhoa(String maSach,String tenSach, String nhaXuatBan, Date ngayXuatBan, Date ngayNhapKho, int soLanTaiBan,

int soLuong) {

this.maSach = maSach;

this.tenSach=tenSach;

this.nhaXuatBan = nhaXuatBan;

this.ngayXuatBan = ngayXuatBan;

this.ngayNhapKho = ngayNhapKho;

this.soLanTaiBan = soLanTaiBan;

this.soLuong = soLuong;

}

public SachGiaoKhoa(String maSach) {

this.maSach = maSach;

}

public SachGiaoKhoa() {

}

public String getMaSach() {

return maSach;

}

public void setMaSach(String maSach) {

this.maSach = maSach;

}

public String getNhaXuatBan() {

return nhaXuatBan;

}

public void setTenSach(String tenSach) {

this.tenSach = tenSach;

}

public String getTenSach() {

return tenSach;

}

public void setNhaXuatBan(String nhaXuatBan) {

this.nhaXuatBan = nhaXuatBan;

}

public Date getNgayXuatBan() {

return ngayXuatBan;

}

public void setNgayXuatBan(Date ngayXuatBan) {

this.ngayXuatBan = ngayXuatBan;

}

public Date getNgayNhapKho() {

return ngayNhapKho;

}

public void setNgayNhapKho(Date ngayNhapKho) {

this.ngayNhapKho = ngayNhapKho;

}

public int getSoLanTaiBan() {

return soLanTaiBan;

}

public void setSoLanTaiBan(int soLanTaiBan) {

this.soLanTaiBan = soLanTaiBan;

}

public int getSoLuong() {

return soLuong;

}

public void setSoLuong(int soLuong) {

this.soLuong = soLuong;

}

public void nhap() {

}

public abstract boolean kiemTraSachMoi();

public void chinhsuathongtin() {};

@Override

public boolean equals(Object obj) {

if (this == obj)

return true;

if (obj == null)

return false;

if (getClass() != obj.getClass())

return false;

SachGiaoKhoa other = (SachGiaoKhoa) obj;

return Objects.equals(maSach, other.maSach);

}

@Override

public String toString() {

return "SachGiaoKhoa [maSach:" + maSach+ ", ten sach: "+ tenSach + ", nhaXuatBan: " + nhaXuatBan + ", ngayXuatBan:"

+ ngayXuatBan.toString() + ", ngayNhapKho: " + ngayNhapKho.toString() + ", soLanTaiBan: " + soLanTaiBan

+ ", soLuong=" + soLuong + "]";

}

}

## Class SachGiaoKhoa\_Cap1

package sach;

import java.util.Objects;

import java.util.Scanner;

public class SachGiaoKhoa\_Cap1 extends SachGiaoKhoa {

public SachGiaoKhoa\_Cap1(String maSach,String tenSach, String nhaXuatBan, Date ngayXuatBan, Date ngayNhapKho, int soLanTaiBan,

int soLuong) {

super(maSach,tenSach, nhaXuatBan,ngayXuatBan, ngayNhapKho, soLanTaiBan, soLuong);

}

public SachGiaoKhoa\_Cap1() {

}

public SachGiaoKhoa\_Cap1(String tenSach) {

super(tenSach);

}

@Override

public boolean kiemTraSachMoi() {

if (super.ngayNhapKho.getNam() >= 2022)

return true;

else

return false;

}

@Override

public void nhap() {

Scanner sc = new Scanner(System.in);

System.out.println("Nhập mã sách");

this.setMaSach(sc.nextLine());

System.out.println("Nhập ten sách");

this.setTenSach(sc.nextLine());

System.out.println("");

System.out.println("Nhập nhà xuất bản");

this.setNhaXuatBan(sc.nextLine());

System.out.println("");

System.out.println("Nhập ngày,tháng,năm xuất bản của sách");

this.ngayXuatBan = new Date();

this.ngayXuatBan.nhapDate();

// Ràng Buộc ngày tháng năm của ngày nhập kho phải lớn hơn hoặc = ngày xuất bản

int check;

do {

System.out.println("");

System.out.println("Nhập ngày,tháng,năm nhập kho của sách");

this.ngayNhapKho = new Date();

this.ngayNhapKho.nhapDate();

check = 0;

if (this.ngayNhapKho.getNam() == this.ngayXuatBan.getNam()) {

if (this.ngayNhapKho.getThang() == this.ngayXuatBan.getThang()) {

if (this.ngayNhapKho.getNgay() < this.ngayXuatBan.getNgay()) {

System.out.println("Vui lòng nhập ngày nhập kho lớn hơn hoặc bằng ngày xuất bản !");

check = 1;

}

} else if (this.ngayNhapKho.getThang() < this.ngayXuatBan.getThang()) {

System.out.println("Vui lòng nhập tháng nhập kho lớn hơn hoặc bằng tháng xuất bản !");

check = 1;

}

} else if (this.ngayNhapKho.getNam() < this.ngayXuatBan.getNam()) {

System.out.println("Vui lòng năm ngày nhập kho lớn hơn hoặc bằng năm xuất bản !");

check = 1;

}

} while (check == 1);

System.out.println("");

do {

System.out.println("Nhập số lần tái bản");

this.setSoLanTaiBan(sc.nextInt());

if (this.getSoLanTaiBan() < 0) {

System.out.println("Vui lòng nhập số lần tái bản lớn hơn hoạc bằng 0 !");

}

} while (this.getSoLanTaiBan() < 0);

System.out.println("");

do {

System.out.println("Nhập số lượng sách ");

this.setSoLuong(sc.nextInt());

if (this.getSoLuong() < 0) {

System.out.println("Vui lòng nhập số lượng lớn hơn hặc bằng 0 !");

}

} while (this.getSoLuong() < 0);

}

@Override

public void chinhsuathongtin() {

Scanner sc=new Scanner(System.in);

System.out.println("Nhập nhà xuất bản bạn muốn chỉnh sửa");

this.setNhaXuatBan(sc.nextLine());

System.out.println("Nhập tên sách bạn muốn chỉnh sửa");

this.setTenSach(sc.nextLine());

System.out.println("Nhập loai sách bạn muốn chỉnh sửa");

System.out.println("");

System.out.println("Nhập ngày,tháng,năm xuất bản của sách bạn muốn chỉnh sửa");

this.ngayXuatBan = new Date();

this.ngayXuatBan.nhapDate();

// Ràng Buộc ngày tháng năm của ngày nhập kho phải lớn hơn hoặc = ngày xuất bản

int check;

do {

System.out.println("");

System.out.println("Nhập ngày,tháng,năm nhập kho của sách bạn muốn chỉnh sửa");

this.ngayNhapKho = new Date();

this.ngayNhapKho.nhapDate();

check = 0;

if (this.ngayNhapKho.getNam() == this.ngayXuatBan.getNam()) {

if (this.ngayNhapKho.getThang() == this.ngayXuatBan.getThang()) {

if (this.ngayNhapKho.getNgay() < this.ngayXuatBan.getNgay()) {

System.out.println("Vui lòng nhập ngày nhập kho lớn hơn hoặc bằng ngày xuất bản !");

check = 1;

}

} else if (this.ngayNhapKho.getThang() < this.ngayXuatBan.getThang()) {

System.out.println("Vui lòng nhập tháng nhập kho lớn hơn hoặc bằng tháng xuất bản !");

check = 1;

}

} else if (this.ngayNhapKho.getNam() < this.ngayXuatBan.getNam()) {

System.out.println("Vui lòng năm ngày nhập kho lớn hơn hoặc bằng năm xuất bản !");

check = 1;

}

} while (check == 1);

System.out.println("");

do {

System.out.println("Nhập số lần tái bản bạn muốn chỉnh sửa");

this.setSoLanTaiBan(sc.nextInt());

if (this.getSoLanTaiBan() < 0) {

System.out.println("Vui lòng nhập số lần tái bản lớn hơn hoạc bằng 0 !");

}

} while (this.getSoLanTaiBan() < 0);

System.out.println("");

do {

System.out.println("Nhập số lượng sách bạn muốn chỉnh sửa");

this.setSoLuong(sc.nextInt());

if (this.getSoLuong() < 0) {

System.out.println("Vui lòng nhập số lượng lớn hơn hặc bằng 0 !");

}

} while (this.getSoLuong() < 0);

}

@Override

public String toString() {

return "SachGiaoKhoa\_Cap1 =" +super.toString() ;

}

@Override

public int compareTo(SachGiaoKhoa o) {

return this.getMaSach().compareTo(o.getMaSach());

}

@Override

public boolean equals(Object obj) {

if (this == obj)

return true;

if (!super.equals(obj))

return false;

if (getClass() != obj.getClass())

return false;

SachGiaoKhoa\_Cap1 other = (SachGiaoKhoa\_Cap1) obj;

return tenSach == other.tenSach;

}

}

## Class SachGiaoKhoa\_Cap2

package sach;

import java.util.Scanner;

public class SachGiaoKhoa\_Cap2 extends SachGiaoKhoa {

public SachGiaoKhoa\_Cap2() {

}

public SachGiaoKhoa\_Cap2(String maSach,String tenSach, String nhaXuatBan, Date ngayXuatBan, Date ngayNhapKho, int soLanTaiBan,

int soLuong) {

super(maSach,tenSach, nhaXuatBan,ngayXuatBan, ngayNhapKho, soLanTaiBan, soLuong);

}

public SachGiaoKhoa\_Cap2(String tenSach) {

super(tenSach);

}

public boolean kiemTraSachMoi() {

if (super.ngayNhapKho.getNam() >= 2022)

return true;

else

return false;

}

@Override

public void nhap() {

Scanner sc = new Scanner(System.in);

System.out.println("Nhập mã sách");

this.setMaSach(sc.nextLine());

System.out.println("Nhập ten sách");

this.setTenSach(sc.nextLine());

System.out.println("");

System.out.println("Nhập nhà xuất bản");

this.setNhaXuatBan(sc.nextLine());

System.out.println("");

System.out.println("Nhập ngày,tháng,năm xuất bản của sách");

this.ngayXuatBan = new Date();

this.ngayXuatBan.nhapDate();

// Ràng Buộc ngày tháng năm của ngày nhập kho phải lớn hơn hoặc = ngày xuất bản

int check;

do {

System.out.println("");

System.out.println("Nhập ngày,tháng,năm nhập kho của sách");

this.ngayNhapKho = new Date();

this.ngayNhapKho.nhapDate();

check = 0;

if (this.ngayNhapKho.getNam() == this.ngayXuatBan.getNam()) {

if (this.ngayNhapKho.getThang() == this.ngayXuatBan.getThang()) {

if (this.ngayNhapKho.getNgay() < this.ngayXuatBan.getNgay()) {

System.out.println("Vui lòng nhập ngày nhập kho lớn hơn hoặc bằng ngày xuất bản !");

check = 1;

}

} else if (this.ngayNhapKho.getThang() < this.ngayXuatBan.getThang()) {

System.out.println("Vui lòng nhập tháng nhập kho lớn hơn hoặc bằng tháng xuất bản !");

check = 1;

}

} else if (this.ngayNhapKho.getNam() < this.ngayXuatBan.getNam()) {

System.out.println("Vui lòng năm ngày nhập kho lớn hơn hoặc bằng năm xuất bản !");

check = 1;

}

} while (check == 1);

System.out.println("");

do {

System.out.println("Nhập số lần tái bản");

this.setSoLanTaiBan(sc.nextInt());

if (this.getSoLanTaiBan() < 0) {

System.out.println("Vui lòng nhập số lần tái bản lớn hơn hoạc bằng 0 !");

}

} while (this.getSoLanTaiBan() < 0);

System.out.println("");

do {

System.out.println("Nhập số lượng sách ");

this.setSoLuong(sc.nextInt());

if (this.getSoLuong() < 0) {

System.out.println("Vui lòng nhập số lượng lớn hơn hặc bằng 0 !");

}

} while (this.getSoLuong() < 0);

}

@Override

public void chinhsuathongtin() {

Scanner sc=new Scanner(System.in);

System.out.println("Nhập nhà xuất bản bạn muốn chỉnh sửa");

this.setNhaXuatBan(sc.nextLine());

System.out.println("Nhập tên sách bạn muốn chỉnh sửa");

this.setTenSach(sc.nextLine());

System.out.println("Nhập loai sách bạn muốn chỉnh sửa");

System.out.println("");

System.out.println("Nhập ngày,tháng,năm xuất bản của sách bạn muốn chỉnh sửa");

this.ngayXuatBan = new Date();

this.ngayXuatBan.nhapDate();

// Ràng Buộc ngày tháng năm của ngày nhập kho phải lớn hơn hoặc = ngày xuất bản

int check;

do {

System.out.println("");

System.out.println("Nhập ngày,tháng,năm nhập kho của sách bạn muốn chỉnh sửa");

this.ngayNhapKho = new Date();

this.ngayNhapKho.nhapDate();

check = 0;

if (this.ngayNhapKho.getNam() == this.ngayXuatBan.getNam()) {

if (this.ngayNhapKho.getThang() == this.ngayXuatBan.getThang()) {

if (this.ngayNhapKho.getNgay() < this.ngayXuatBan.getNgay()) {

System.out.println("Vui lòng nhập ngày nhập kho lớn hơn hoặc bằng ngày xuất bản !");

check = 1;

}

} else if (this.ngayNhapKho.getThang() < this.ngayXuatBan.getThang()) {

System.out.println("Vui lòng nhập tháng nhập kho lớn hơn hoặc bằng tháng xuất bản !");

check = 1;

}

} else if (this.ngayNhapKho.getNam() < this.ngayXuatBan.getNam()) {

System.out.println("Vui lòng năm ngày nhập kho lớn hơn hoặc bằng năm xuất bản !");

check = 1;

}

} while (check == 1);

System.out.println("");

do {

System.out.println("Nhập số lần tái bản bạn muốn chỉnh sửa");

this.setSoLanTaiBan(sc.nextInt());

if (this.getSoLanTaiBan() < 0) {

System.out.println("Vui lòng nhập số lần tái bản lớn hơn hoạc bằng 0 !");

}

} while (this.getSoLanTaiBan() < 0);

System.out.println("");

do {

System.out.println("Nhập số lượng sách bạn muốn chỉnh sửa");

this.setSoLuong(sc.nextInt());

if (this.getSoLuong() < 0) {

System.out.println("Vui lòng nhập số lượng lớn hơn hặc bằng 0 !");

}

} while (this.getSoLuong() < 0);

}

@Override

public String toString() {

return "SachGiaoKhoa\_Cap2 "+ super.toString() ;

}

@Override

public int compareTo(SachGiaoKhoa o) {

return this.getMaSach().compareTo(o.maSach);

}

@Override

public boolean equals(Object obj) {

if (this == obj)

return true;

if (!super.equals(obj))

return false;

if (getClass() != obj.getClass())

return false;

SachGiaoKhoa\_Cap2 other = (SachGiaoKhoa\_Cap2) obj;

return tenSach == other.tenSach;

}

}

## Class SachGiaoKhoa\_Cap3

package sach;

import java.util.Objects;

import java.util.Scanner;

public class SachGiaoKhoa\_Cap3 extends SachGiaoKhoa {

public SachGiaoKhoa\_Cap3(String maSach,String tenSach, String nhaXuatBan, Date ngayXuatBan, Date ngayNhapKho, int soLanTaiBan,

int soLuong) {

super(maSach,tenSach, nhaXuatBan,ngayXuatBan, ngayNhapKho, soLanTaiBan, soLuong);

}

public SachGiaoKhoa\_Cap3() {

}

public SachGiaoKhoa\_Cap3(String tenSach) {

super(tenSach);

}

public boolean kiemTraSachMoi() {

if (super.ngayNhapKho.getNam() >= 2022)

return true;

else

return false;

}

@Override

public void nhap() {

Scanner sc = new Scanner(System.in);

System.out.println("Nhập mã sách");

this.setMaSach(sc.nextLine());

System.out.println("Nhập ten sách");

this.setTenSach(sc.nextLine());

System.out.println("");

System.out.println("Nhập nhà xuất bản");

this.setNhaXuatBan(sc.nextLine());

System.out.println("");

System.out.println("Nhập ngày,tháng,năm xuất bản của sách");

this.ngayXuatBan = new Date();

this.ngayXuatBan.nhapDate();

// Ràng Buộc ngày tháng năm của ngày nhập kho phải lớn hơn hoặc = ngày xuất bản

int check;

do {

System.out.println("");

System.out.println("Nhập ngày,tháng,năm nhập kho của sách");

this.ngayNhapKho = new Date();

this.ngayNhapKho.nhapDate();

check = 0;

if (this.ngayNhapKho.getNam() == this.ngayXuatBan.getNam()) {

if (this.ngayNhapKho.getThang() == this.ngayXuatBan.getThang()) {

if (this.ngayNhapKho.getNgay() < this.ngayXuatBan.getNgay()) {

System.out.println("Vui lòng nhập ngày nhập kho lớn hơn hoặc bằng ngày xuất bản !");

check = 1;

}

} else if (this.ngayNhapKho.getThang() < this.ngayXuatBan.getThang()) {

System.out.println("Vui lòng nhập tháng nhập kho lớn hơn hoặc bằng tháng xuất bản !");

check = 1;

}

} else if (this.ngayNhapKho.getNam() < this.ngayXuatBan.getNam()) {

System.out.println("Vui lòng năm ngày nhập kho lớn hơn hoặc bằng năm xuất bản !");

check = 1;

}

} while (check == 1);

System.out.println("");

do {

System.out.println("Nhập số lần tái bản");

this.setSoLanTaiBan(sc.nextInt());

if (this.getSoLanTaiBan() < 0) {

System.out.println("Vui lòng nhập số lần tái bản lớn hơn hoạc bằng 0 !");

}

} while (this.getSoLanTaiBan() < 0);

System.out.println("");

do {

System.out.println("Nhập số lượng sách ");

this.setSoLuong(sc.nextInt());

if (this.getSoLuong() < 0) {

System.out.println("Vui lòng nhập số lượng lớn hơn hặc bằng 0 !");

}

} while (this.getSoLuong() < 0);

}

@Override

public void chinhsuathongtin() {

Scanner sc=new Scanner(System.in);

System.out.println("Nhập nhà xuất bản bạn muốn chỉnh sửa");

this.setNhaXuatBan(sc.nextLine());

System.out.println("Nhập tên sách bạn muốn chỉnh sửa");

this.setTenSach(sc.nextLine());

System.out.println("Nhập loai sách bạn muốn chỉnh sửa");

System.out.println("");

System.out.println("Nhập ngày,tháng,năm xuất bản của sách bạn muốn chỉnh sửa");

this.ngayXuatBan = new Date();

this.ngayXuatBan.nhapDate();

// Ràng Buộc ngày tháng năm của ngày nhập kho phải lớn hơn hoặc = ngày xuất bản

int check;

do {

System.out.println("");

System.out.println("Nhập ngày,tháng,năm nhập kho của sách bạn muốn chỉnh sửa");

this.ngayNhapKho = new Date();

this.ngayNhapKho.nhapDate();

check = 0;

if (this.ngayNhapKho.getNam() == this.ngayXuatBan.getNam()) {

if (this.ngayNhapKho.getThang() == this.ngayXuatBan.getThang()) {

if (this.ngayNhapKho.getNgay() < this.ngayXuatBan.getNgay()) {

System.out.println("Vui lòng nhập ngày nhập kho lớn hơn hoặc bằng ngày xuất bản !");

check = 1;

}

} else if (this.ngayNhapKho.getThang() < this.ngayXuatBan.getThang()) {

System.out.println("Vui lòng nhập tháng nhập kho lớn hơn hoặc bằng tháng xuất bản !");

check = 1;

}

} else if (this.ngayNhapKho.getNam() < this.ngayXuatBan.getNam()) {

System.out.println("Vui lòng năm ngày nhập kho lớn hơn hoặc bằng năm xuất bản !");

check = 1;

}

} while (check == 1);

System.out.println("");

do {

System.out.println("Nhập số lần tái bản bạn muốn chỉnh sửa");

this.setSoLanTaiBan(sc.nextInt());

if (this.getSoLanTaiBan() < 0) {

System.out.println("Vui lòng nhập số lần tái bản lớn hơn hoạc bằng 0 !");

}

} while (this.getSoLanTaiBan() < 0);

System.out.println("");

do {

System.out.println("Nhập số lượng sách bạn muốn chỉnh sửa");

this.setSoLuong(sc.nextInt());

if (this.getSoLuong() < 0) {

System.out.println("Vui lòng nhập số lượng lớn hơn hặc bằng 0 !");

}

} while (this.getSoLuong() < 0);

}

@Override

public String toString() {

return "SachGiaoKhoa\_Cap3 "+ super.toString();

}

@Override

public int compareTo(SachGiaoKhoa o) {

return this.getMaSach().compareTo(o.getMaSach());

}

@Override

public boolean equals(Object obj) {

if (this == obj)

return true;

if (!super.equals(obj))

return false;

if (getClass() != obj.getClass())

return false;

SachGiaoKhoa\_Cap3 other = (SachGiaoKhoa\_Cap3) obj;

return tenSach == other.tenSach;

}

}

## Class QuanLyKhoSach

package sach;

import java.io.File;

import java.io.FileInputStream;

import java.io.FileOutputStream;

import java.io.ObjectInputStream;

import java.io.ObjectOutputStream;

import java.io.OutputStream;

import java.util.ArrayList;

import java.util.Collection;

import java.util.Collections;

import java.util.Comparator;

import java.util.Iterator;

public class QuanLyKhoSach {

private ArrayList<SachGiaoKhoa> list;

SachGiaoKhoa sgk;

public QuanLyKhoSach() {

this.list = new ArrayList<SachGiaoKhoa>();

}

public QuanLyKhoSach(ArrayList<SachGiaoKhoa> list) {

this.list = list;

}

public void themVaoDanhSach(SachGiaoKhoa sgk) {

if(!kiemTraTonTai\_TheoMaSach(sgk.getMaSach())) {

this.list.add(sgk);

}else {

System.out.println("Đã Trùng Mã Sách , Không Thể Vào Danh Sách !");

return;

}

}

public void inDanhSach() {

for (SachGiaoKhoa sgk : list) {

if (sgk instanceof SachGiaoKhoa)

System.out.println(sgk);

}

}

public boolean kiemTraDanhSachRong() {

return this.list.isEmpty();

}

public int soLuongSachHienCo() {

return this.list.size();

}

public int soluongSachGiaoKhoa\_Cap1() {

int count = 0;

for (SachGiaoKhoa sgk : list) {

if (sgk instanceof SachGiaoKhoa\_Cap1)

count++;

}

return count;

}

public int soluongSachGiaoKhoa\_Cap2() {

int count = 0;

for (SachGiaoKhoa sgk : list) {

if (sgk instanceof SachGiaoKhoa\_Cap2)

count++;

}

return count;

}

public int soluongSachGiaoKhoa\_Cap3() {

int count = 0;

for (SachGiaoKhoa sgk : list) {

if (sgk instanceof SachGiaoKhoa\_Cap3)

count++;

}

return count;

}

public void lamRongDanhSach() {

if (this.list.size() == 0)

System.out.println("Danh sách đã rỗng sẵn không thể làm rỗng !");

else {

this.list.removeAll(list);

System.out.println("Đã làm rỗng !");

}

}

public boolean kiemTraTonTai\_TheoMaSach(String maSach) {

if (this.list.size() == 0) {

System.out.println("Danh sách đã rỗng không thể kiểm tra sự tồn tại");

return false;

}

for (SachGiaoKhoa sgk : list) {

if (sgk.getMaSach().equals(maSach)) {

return true;

}

}

return false;

}

public void timSach\_TheoMaSach(String maSach) {

if (this.list.size() == 0) {

System.out.println("Danh sách đã rỗng không thể tìm sách");

return;

}

for (SachGiaoKhoa sgk : list) {

if (sgk.getMaSach().equals(maSach)) {

System.out.println(sgk);

}

}

}

public void timSach\_TheoTen(String tenSach) {

if(this.list.size()==0) {

System.out.println("Danh sách đã rỗng không thể tìm sách");

return ;

}

for(SachGiaoKhoa sgk :list) {

if(sgk.getTenSach().equalsIgnoreCase(tenSach)) {

System.out.println(sgk);

}

}

}

public void xoaSach\_TheoMaSach(String maSach) {

if (this.list.size() == 0) {

System.out.println("Danh sách đã rỗng không thể xóa sách");

return;

}

Iterator<SachGiaoKhoa> iterator = this.list.iterator();

while (iterator.hasNext()) {

SachGiaoKhoa sgk = iterator.next();

if (sgk.getMaSach().equals(maSach)) {

iterator.remove();

System.out.println("Đã xóa sách có mã " + maSach);

return;

}

}

System.out.println("Không tìm thấy sách có mã " + maSach + " trong danh sách");

}

public void xoaSach\_TheoTenSach(String tenSach) {

for(int i=0;i<this.list.size();i++) {

SachGiaoKhoa sgk=this.list.get(i);

if(sgk.getTenSach().equalsIgnoreCase(tenSach)) {

System.out.println("da xoa sach co ma :"+sgk.getTenSach());

this.list.remove(i);

}

}

}

public void sapXepTangDan\_TheoSoLuong() {

Collections.sort(this.list, new Comparator<SachGiaoKhoa>() {

@Override

public int compare(SachGiaoKhoa o1, SachGiaoKhoa o2) {

if (o1.getSoLuong() < o2.getSoLuong())

return -1;

else if (o1.getSoLuong() > o2.getSoLuong())

return 1;

else

return 0;

}

});

}

public void sapXepGiamDan\_TheoSoLuong() {

Collections.sort(this.list, new Comparator<SachGiaoKhoa>() {

@Override

public int compare(SachGiaoKhoa o1, SachGiaoKhoa o2) {

if (o1.getSoLuong() < o2.getSoLuong())

return 1;

else if (o1.getSoLuong() > o2.getSoLuong())

return -1;

else

return 0;

}

});

}

public void sapXepTangDan\_TheoMaSach() {

Collections.sort(this.list);

}

public void sapXepGiamDan\_TheoMaSach() {

Collections.sort(this.list, new Comparator<SachGiaoKhoa>() {

@Override

public int compare(SachGiaoKhoa o1, SachGiaoKhoa o2) {

if (o1.getMaSach().compareTo(o2.getMaSach()) > 0) {

return -1;

} else if (o1.getMaSach().compareTo(o2.getMaSach()) < 0) {

return 1;

} else {

return 0;

}

}

});

}

public void inSachCap1() {

for (SachGiaoKhoa sgk : list) {

if (sgk instanceof SachGiaoKhoa\_Cap1) {

System.out.println(sgk);

}

}

}

public void inSachCa2() {

for (SachGiaoKhoa sgk : list) {

if (sgk instanceof SachGiaoKhoa\_Cap2) {

System.out.println(sgk);

}

}

}

public void inSachCap3() {

for (SachGiaoKhoa sgk : list) {

if (sgk instanceof SachGiaoKhoa\_Cap3) {

System.out.println(sgk);

}

}

}

public void luuFile(File f) {

try {

OutputStream os=new FileOutputStream(f);

ObjectOutputStream oos=new ObjectOutputStream(os);

for (SachGiaoKhoa sgk : list) {

oos.writeObject(sgk);

}

System.out.println("Đã ghi file vào "+f.getAbsolutePath());

oos.flush();

oos.close();

} catch (Exception e) {

e.printStackTrace();

}

}

public void docFile(File f) {

try {

ObjectInputStream ois=new ObjectInputStream(new FileInputStream(f));

SachGiaoKhoa sgk;

while(true) {

try {

sgk=(SachGiaoKhoa)ois.readObject();

this.list.add(sgk);

} catch (Exception e) {

break;

}

}

System.out.println("Đã đọc file "+f.getAbsolutePath());

ois.close();

} catch (Exception e) {

e.printStackTrace();

}

}

public boolean KiemTraSachMoi\_TheoMaSach(String maSach) {

for (SachGiaoKhoa sgk : list) {

if(sgk.getMaSach().equals(maSach)) {

return sgk.kiemTraSachMoi();

}

}

System.out.println("Không có sách mà bạn đã nhập");

return false;

}

public void chinhSuaThongTin\_TheoMaSach(String maSach) {

for(int i=0;i<this.list.size();i++) {

SachGiaoKhoa sgk=this.list.get(i);

if(sgk.getMaSach().equals(maSach)) {

if(sgk instanceof SachGiaoKhoa\_Cap1) {

SachGiaoKhoa\_Cap1 sgk\_c1=(SachGiaoKhoa\_Cap1)this.list.get(i);

sgk\_c1.chinhsuathongtin();

this.list.set(i, sgk\_c1);

}else if(sgk instanceof SachGiaoKhoa\_Cap2) {

SachGiaoKhoa\_Cap2 sgk\_c2=(SachGiaoKhoa\_Cap2)this.list.get(i);

sgk\_c2.chinhsuathongtin();

this.list.set(i, sgk\_c2);

}else if(sgk instanceof SachGiaoKhoa\_Cap3) {

SachGiaoKhoa\_Cap3 sgk\_c3=(SachGiaoKhoa\_Cap3)this.list.get(i);

sgk\_c3.chinhsuathongtin();

this.list.set(i, sgk\_c3);

}

}

}

}

}

## Class Menu

package sach;

import java.io.File;

import java.util.Scanner;

import sach.QuanLyKhoSach;

import sach.SachGiaoKhoa;

import sach.SachGiaoKhoa\_Cap1;

import sach.SachGiaoKhoa\_Cap2;

import sach.SachGiaoKhoa\_Cap3;

public class Menu {

Scanner sc = new Scanner(System.in);

QuanLyKhoSach list = new QuanLyKhoSach();

SachGiaoKhoa sgk;

SachGiaoKhoa\_Cap1 sgkc1;

SachGiaoKhoa\_Cap2 sgkc2;

SachGiaoKhoa\_Cap3 sgkc3;

public void menu() {

int chon = 0;

do {

System.out.println("");

System.out.println("------Menu------");

System.out.println("1. Thêm Sách Giáo Khoa Vào Danh Sách");

System.out.println("2. In Danh Sách");

System.out.println("3. In Sách Giáo Khoa Cấp 1");

System.out.println("4. In Sách Giáo Khoa Cấp 2");

System.out.println("5. In Sách Giáo Khoa Cấp 3");

System.out.println("6. Kiểm Tra Danh Sách Rỗng");

System.out.println("7. Số Lượng Sách Giáo Khoa Hiện Có");

System.out.println("8. Số Lượng Sách Giáo Khoa Cấp 1");

System.out.println("9. Số Lượng Sách Giáo Khoa Cấp 2");

System.out.println("10.Số Lượng Sách Giáo Khoa Cấp 3");

System.out.println("11.Làm Rỗng Danh Sách");

System.out.println("12.Kiểm Tra Tồn Tại ( Theo Mã Sách)");

System.out.println("13.Tìm Sách (Theo Mã Sách)");

System.out.println("14.Tìm Sách (Theo Tên Sách)");

System.out.println("15.Xóa Sách (Theo Mã Sách)");

System.out.println("16.Sắp Xếp Tăng Dần (Theo Số Lượng)");

System.out.println("17.Sắp Xếp Giảm Dần (Theo Số Lượng)");

System.out.println("18.Sắp Xếp Tăng Dần (Theo Mã Sách)");

System.out.println("19.Sắp Xếp Giảm Dần (Theo Mã Sách)");

System.out.println("20.Xóa Sách ( Theo Tên Sách)");

System.out.println("21.Chỉnh Sửa Thông Tin (Theo Mã Sách )");

System.out.println("22.Lưu File");

System.out.println("23.Đọc File");

System.out.println("24.Kiểm tra sách mới (số năm lớn hơn 2021)");

System.out.println("0. Thoát");

System.out.println("");

chon = sc.nextInt();

sc.nextLine();

if (chon == 1) {

int luaChon = 0;

// Nếu Lựa Chọn khác 1,2,3 thi nhập lại

do {

System.out.println("---Hãy Chọn Sách Giáo Khoa---");

System.out.println("1.Sách Giáo Khoa Cấp 1");

System.out.println("2.Sách Giáo Khoa Cấp 2");

System.out.println("3.Sách Giáo Khoa Cấp 3");

luaChon = sc.nextInt();

if (luaChon == 1) {

sgkc1 = new SachGiaoKhoa\_Cap1();

sgkc1.nhap();

this.list.themVaoDanhSach(sgkc1);

} else if (luaChon == 2) {

sgkc2 = new SachGiaoKhoa\_Cap2();

sgkc2.nhap();

this.list.themVaoDanhSach(sgkc2);

} else if (luaChon == 3) {

sgkc3 = new SachGiaoKhoa\_Cap3();

sgkc3.nhap();

this.list.themVaoDanhSach(sgkc3);

} else {

System.out.println("Vui Lòng Chọn Sách Hợp Lệ !");

}

} while (luaChon != 1 && luaChon != 2 && luaChon != 3);

} else if (chon == 2) {

this.list.inDanhSach();

} else if (chon == 3) {

this.list.inSachCap1();

} else if (chon == 4) {

this.list.inSachCa2();

} else if (chon == 5) {

this.list.inSachCap3();

} else if (chon == 6) {

System.out.println("Danh Sách Có Rỗng Không : " + this.list.kiemTraDanhSachRong());

} else if (chon == 7) {

System.out.println("Số Lượng Sách Hiện Có : " + this.list.soLuongSachHienCo());

} else if (chon == 8) {

System.out.println("Số Lượng Sách Giáo Khóa Cấp 1 : " + this.list.soluongSachGiaoKhoa\_Cap1());

} else if (chon == 9) {

System.out.println("Số Lượng Sách Giáo Khóa Cấp 2 : " + this.list.soluongSachGiaoKhoa\_Cap2());

} else if (chon == 10) {

System.out.println("Số Lượng Sách Giáo Khóa Cấp 3 : " + this.list.soluongSachGiaoKhoa\_Cap3());

} else if (chon == 11) {

this.list.lamRongDanhSach();

} else if (chon == 12) {

System.out.println("Nhập Mã Sách Bạn Muốn Kiểm Tra");

String maSach = sc.nextLine();

System.out.println("Kiểm Tra Tồn Tại Theo Mã Sách : " + this.list.kiemTraTonTai\_TheoMaSach(maSach));

} else if (chon == 13) {

System.out.println("Nhập Mã Sách Bạn Muốn Tìm Kiếm");

String maSach = sc.nextLine();

this.list.timSach\_TheoMaSach(maSach);

} else if (chon == 14) {

System.out.println("Nhập Tên Sách Bạn Muốn Tìm");

String tenSach = sc.nextLine();

this.list.timSach\_TheoTen(tenSach);

} else if (chon == 15) {

System.out.println("Nhập Mã Sách Bạn Muốn Xóa");

String maSach = sc.nextLine();

this.list.xoaSach\_TheoMaSach(maSach);

} else if (chon == 16) {

this.list.sapXepTangDan\_TheoSoLuong();

this.list.inDanhSach();

} else if (chon == 17) {

this.list.sapXepGiamDan\_TheoSoLuong();

this.list.inDanhSach();

} else if (chon == 18) {

this.list.sapXepTangDan\_TheoMaSach();

this.list.inDanhSach();

} else if (chon == 19) {

this.list.sapXepGiamDan\_TheoMaSach();

this.list.inDanhSach();

} else if(chon==20) {

System.out.println("Nhập tên sách bạn muốn xóa");

String ten=sc.nextLine();

this.list.xoaSach\_TheoTenSach(ten);

}else if(chon==21) {

System.out.println("Nhập mã sách mà bạn muốn chỉnh sửa thông tin");

String maSach=sc.nextLine();

this.list.chinhSuaThongTin\_TheoMaSach(maSach);

}else if(chon==22) {

System.out.println("Nhập đường dẫn file bạn muốn lưu");

String file=sc.nextLine();

File tenFile=new File(file);

this.list.luuFile(tenFile);

}else if(chon==23) {

System.out.println("Nhập đường dẫn file bạn muốn đọc");

String file=sc.nextLine();

File tenFile=new File(file);

this.list.docFile(tenFile);

}else if(chon==24) {

System.out.println("Nhập mã sách bạn muốn kiểm tra");

String maSach=sc.nextLine();

System.out.println("Sách mới :"+this.list.KiemTraSachMoi\_TheoMaSach(maSach));

}else if(chon==0) {

System.out.println("Bạn Đã Thoát Khỏi Menu Xin Cảm Ơn !");

}else {

System.out.println("Vui lòng chọn trong menu");

}

}while(chon!=0);

}

}

## Class Test

package sach;

public class Test {

public static void main(String[] args) {

Menu menu = new Menu();

menu.menu();

}

}

**V.Kiểm thử**

Bảng thông tin sách giáo khoa cấp 1

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Mã Sách | Tên Sách | Nhà Xuất Bản | Ngày Xuất Bản | Ngày Nhập Kho | Số Lần Tái Bản | Số Lượng |
| 007 | Toán | Nhã Nam | 12/08/2005 | 16/9/2005 | 10 | 1000 |
| 004 | Tiếng Việt | Phương Nam | 18/08/2003 | 20/9/2004 | 20 | 2000 |

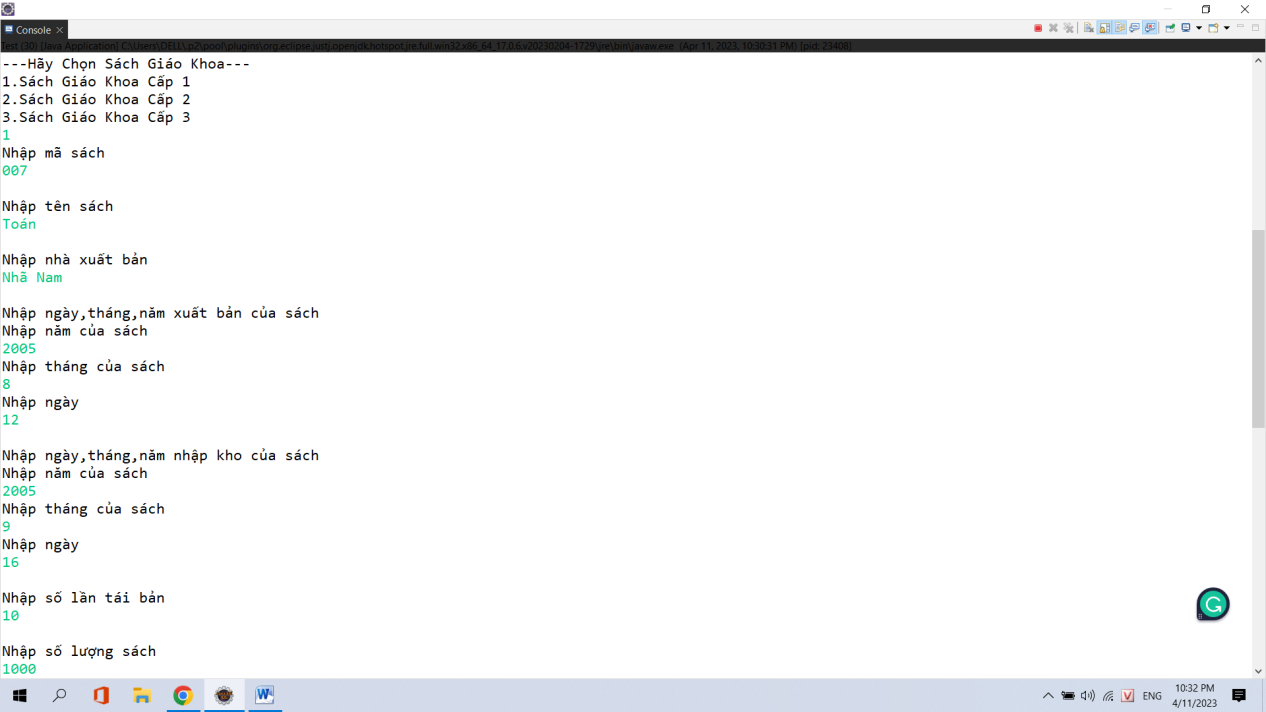
Bảng thông tin sách giáo khoa cấp 2

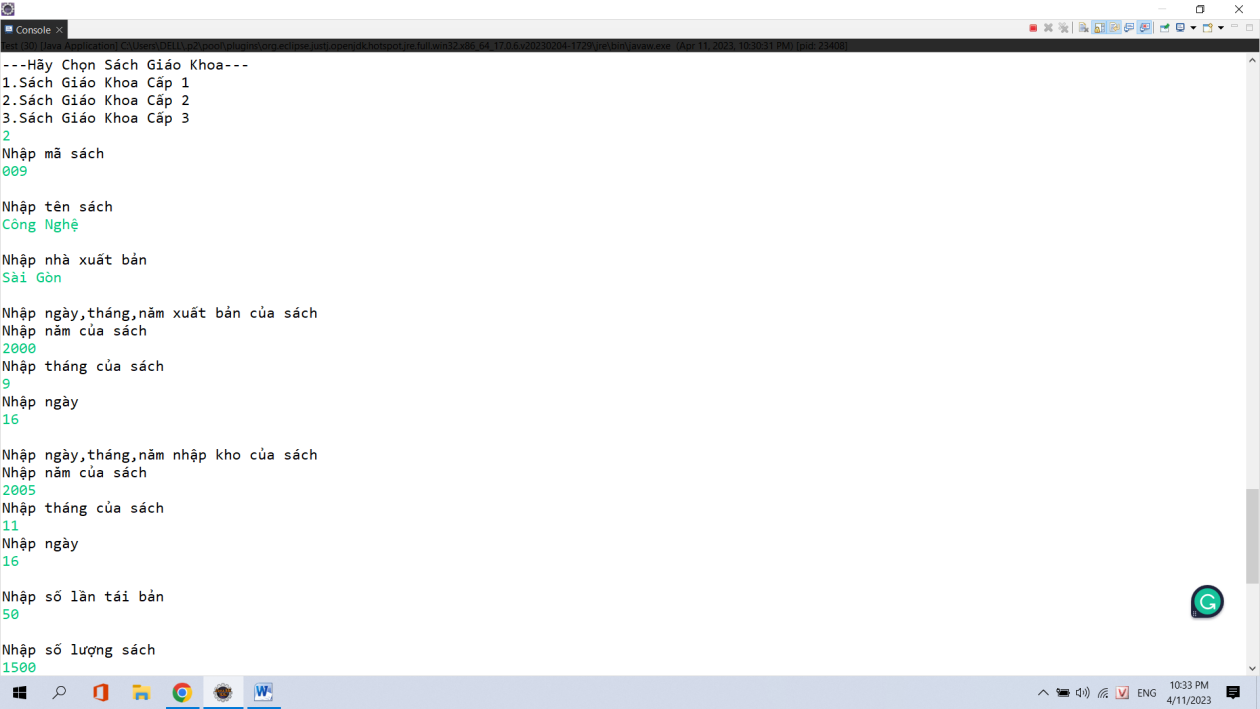
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Mã Sách | Tên Sách | Nhà Xuất Bản | Ngày Xuất Bản | Ngày Nhập Kho | Số Lần Tái Bản | Số Lượng |
| 009 | Công nghệ | Sài Gòn | 16/9/2000 | 16/11/2005 | 50 | 1500 |

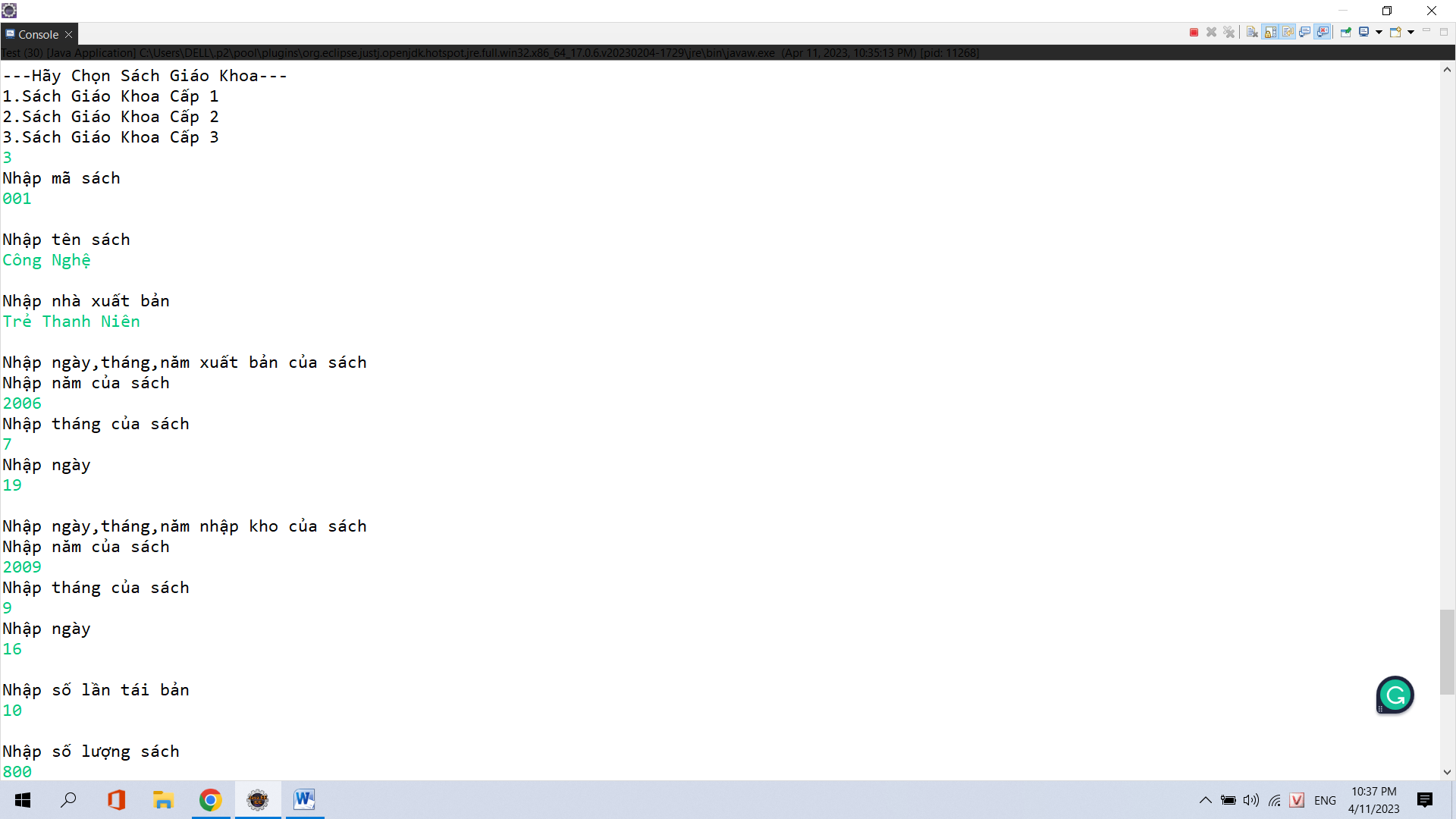
Bảng thông tin sách giáo khoa cấp 3:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Mã Sách | Tên sách | Nhà Xuất Bản | Ngày Xuất Bản | Ngày Nhập Kho | Số Lần Tái Bản | Số Lượng |
| 001 | Công Nghệ | Trẻ Thanh Niên | 19/7/2006 | 16/9/2009 | 10 | 800 |
| 005 | Ngoại Ngữ | Hồ Khanh | 18/9/2005 | 14/5/2009 | 2 | 400 |

1. Nhập thông tin



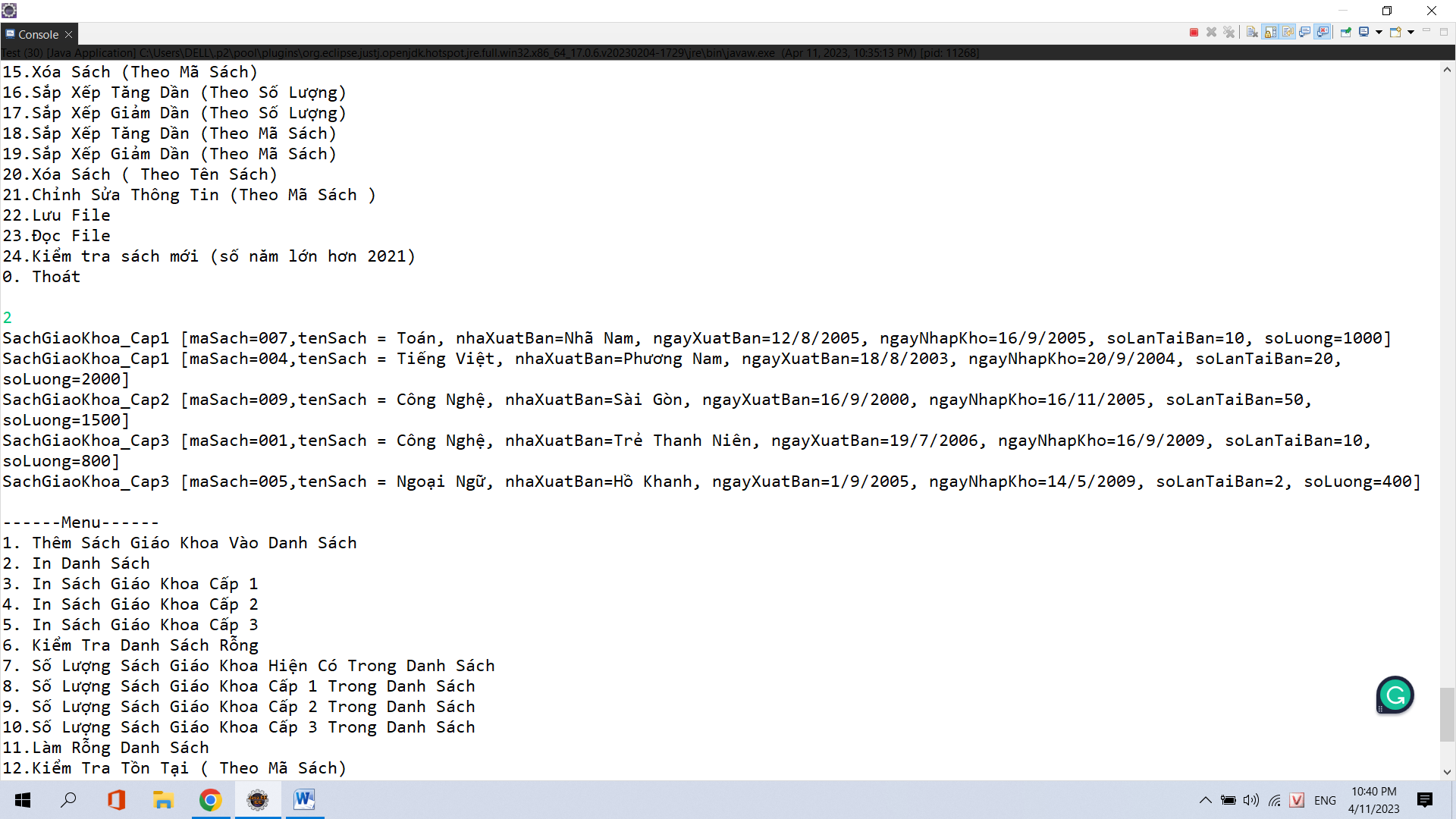




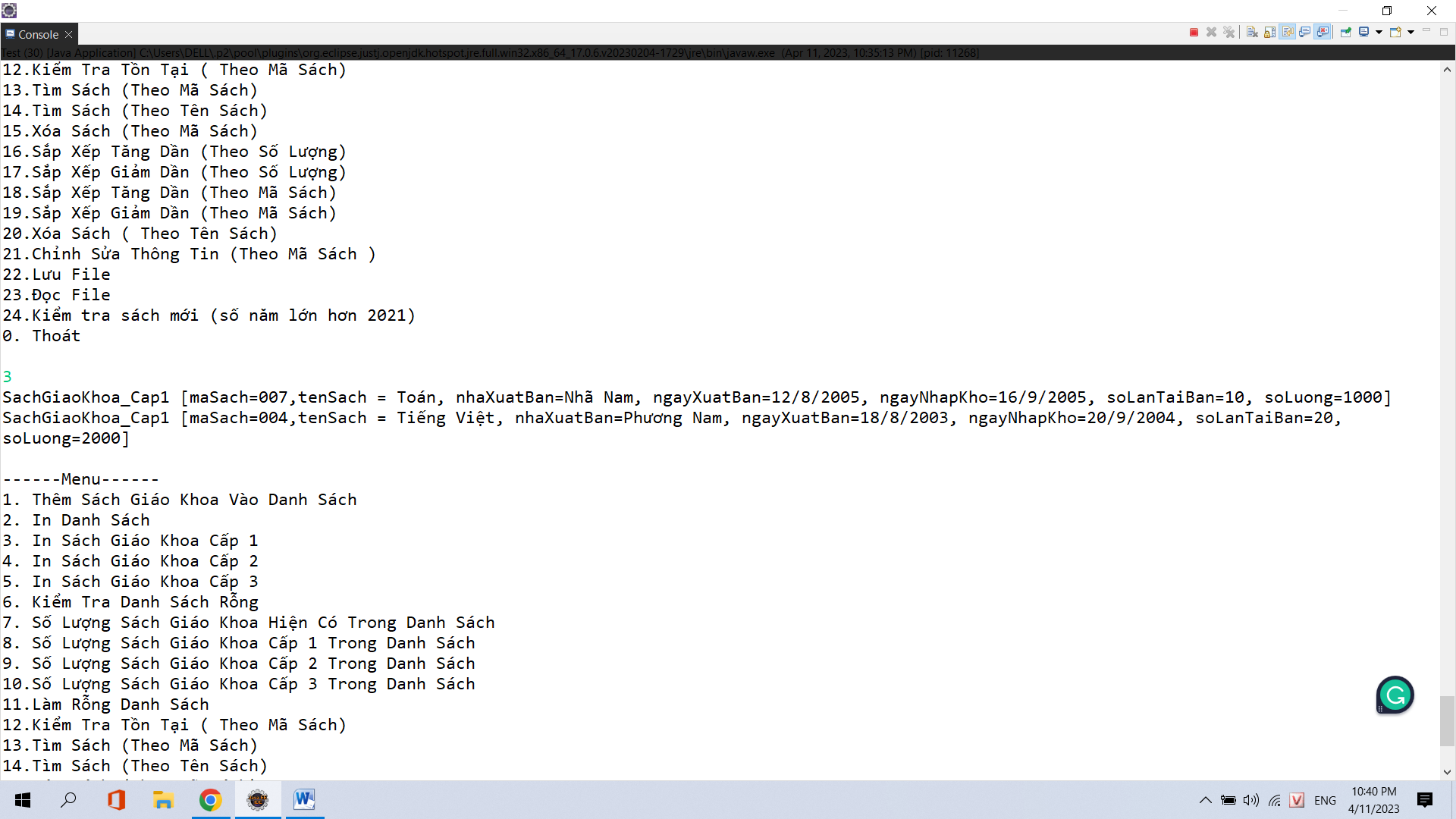




1. Xuất thông tin của kho tổng



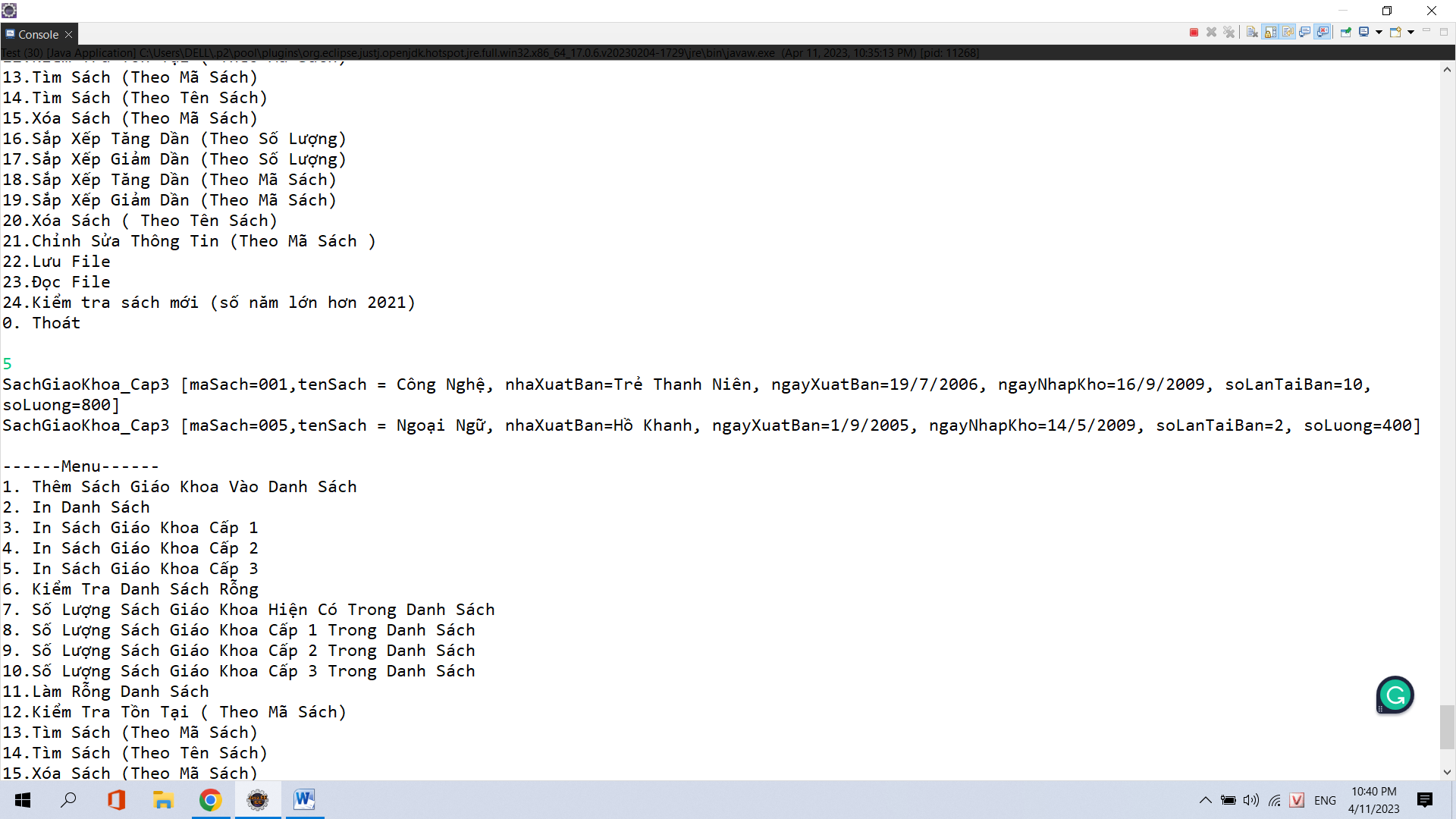
3.Xuất thông tin của kho cấp 1



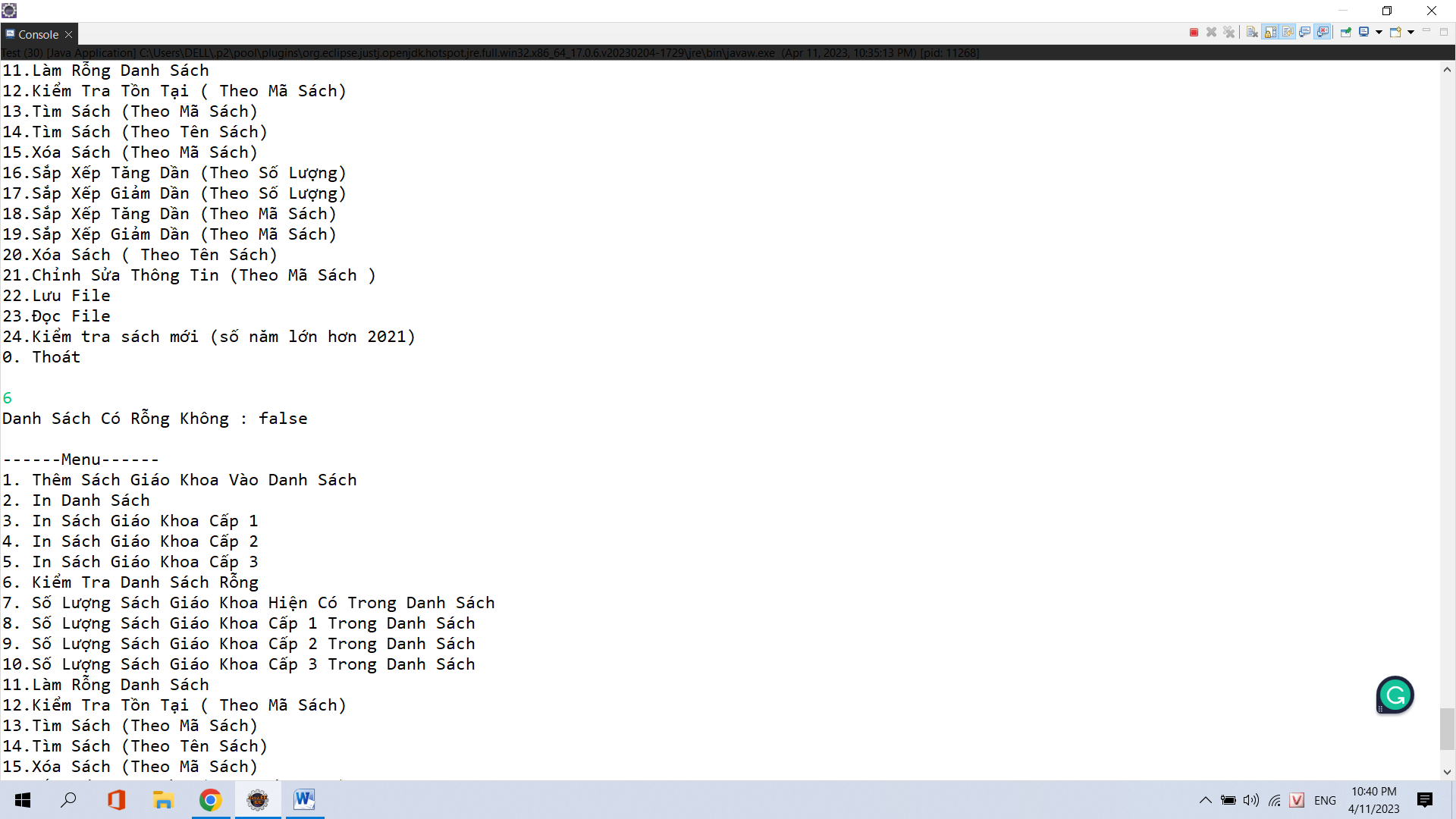
1. Xuất thông tin của kho cấp 2



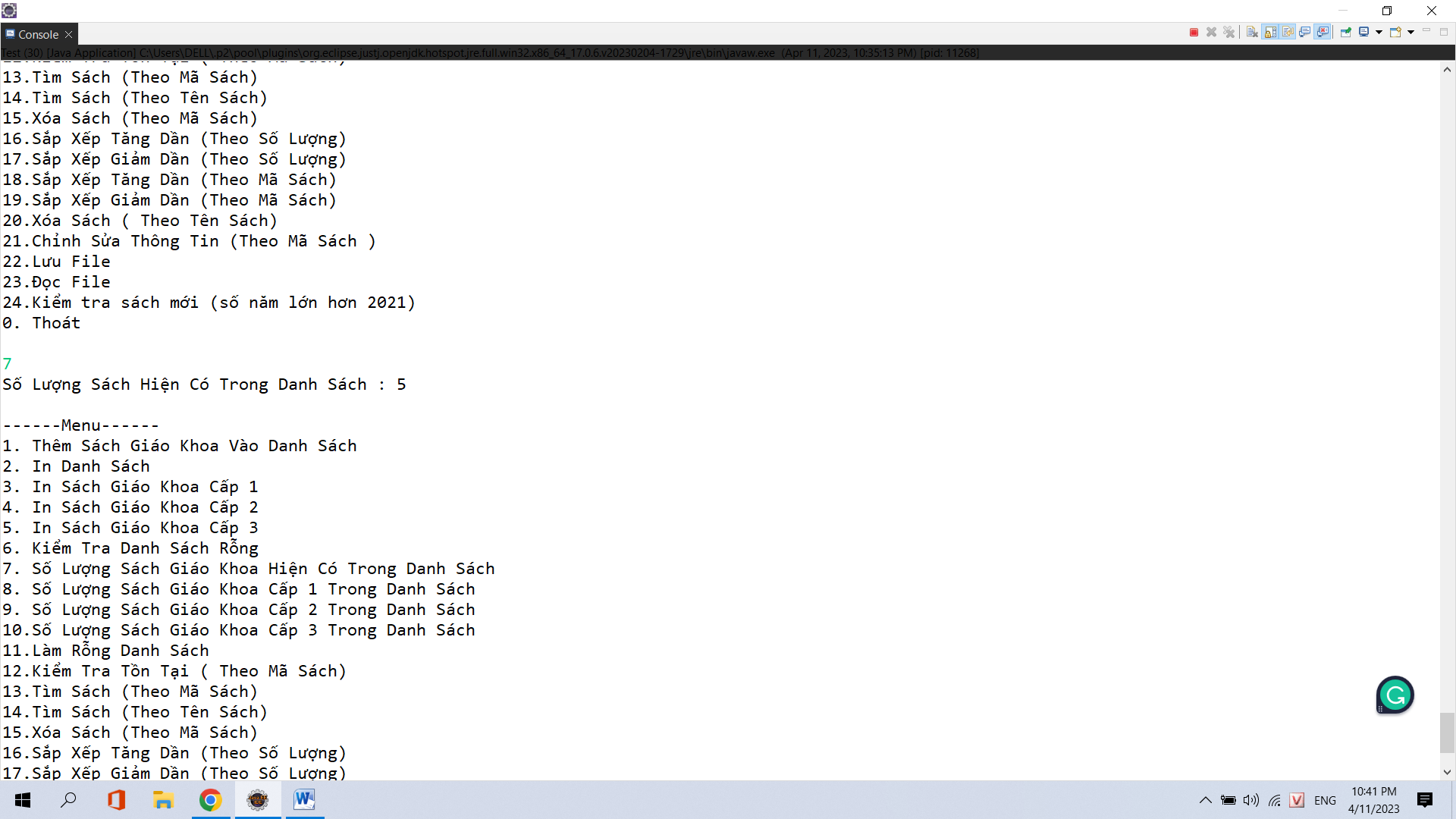
5.xuất thông tin kho cấp 3



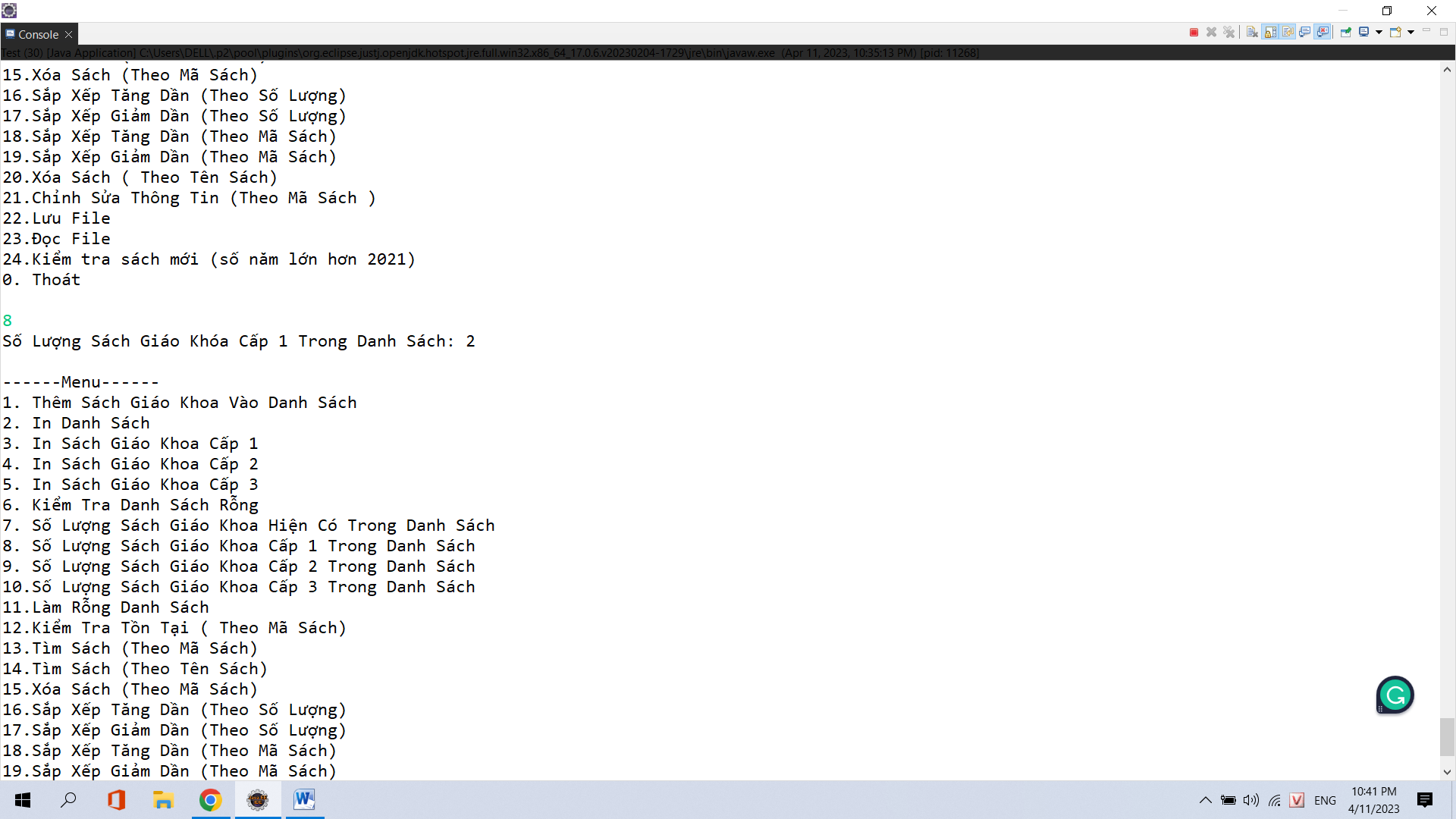
1. Kiểm tra kho có rỗng không

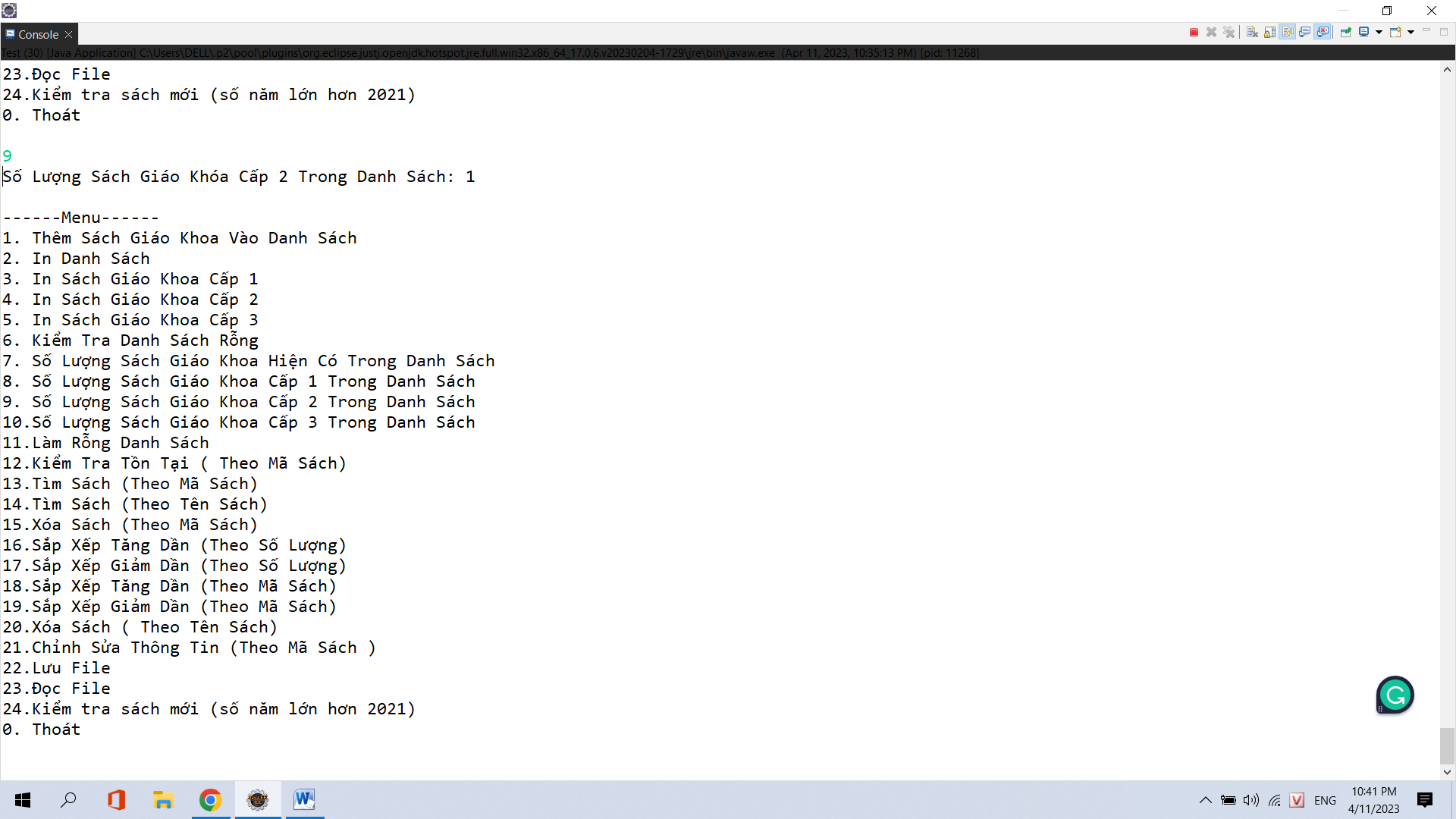


7.xuất ra số lượng tổng



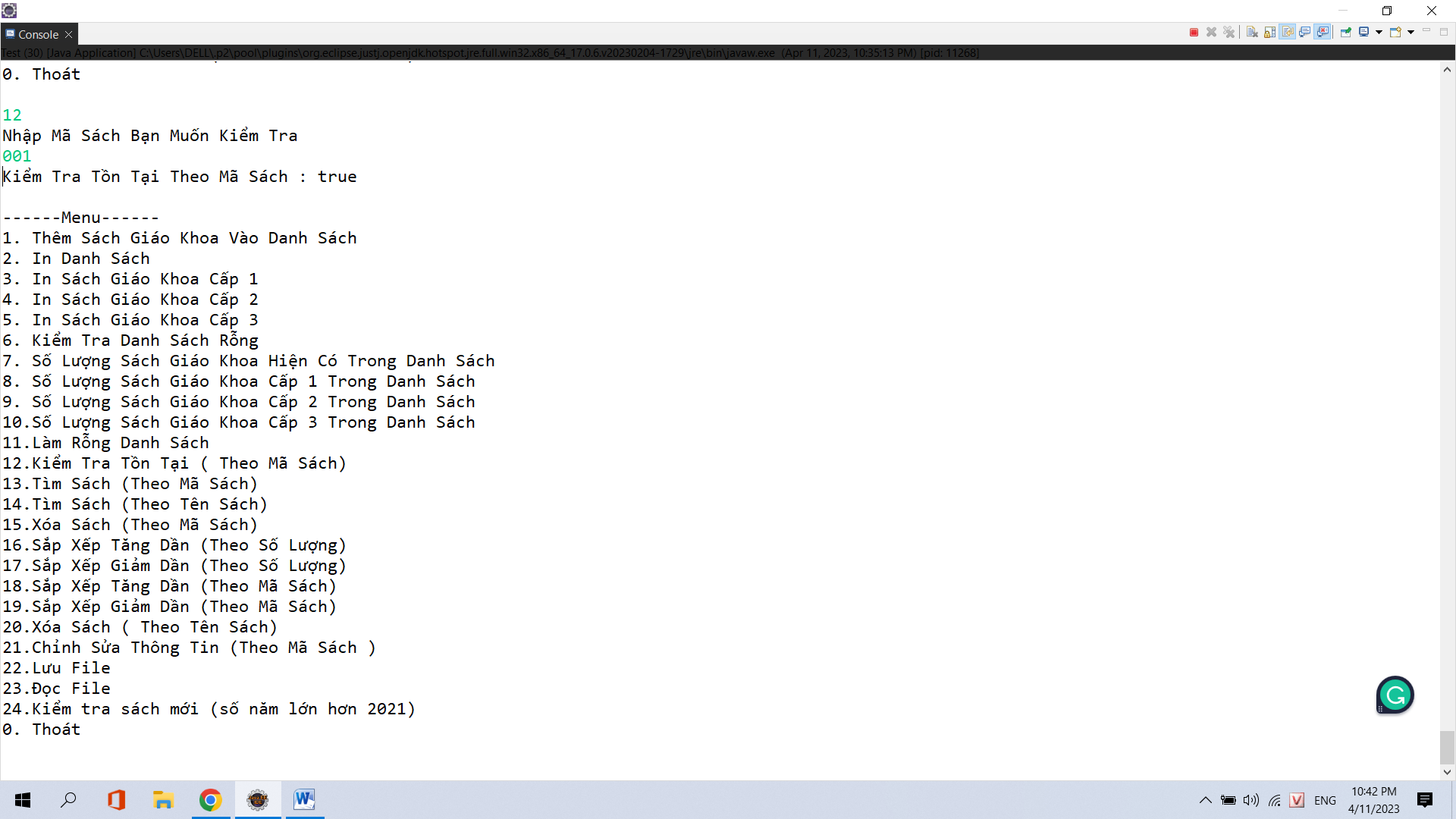
8.xuất ra số lượng từng loại



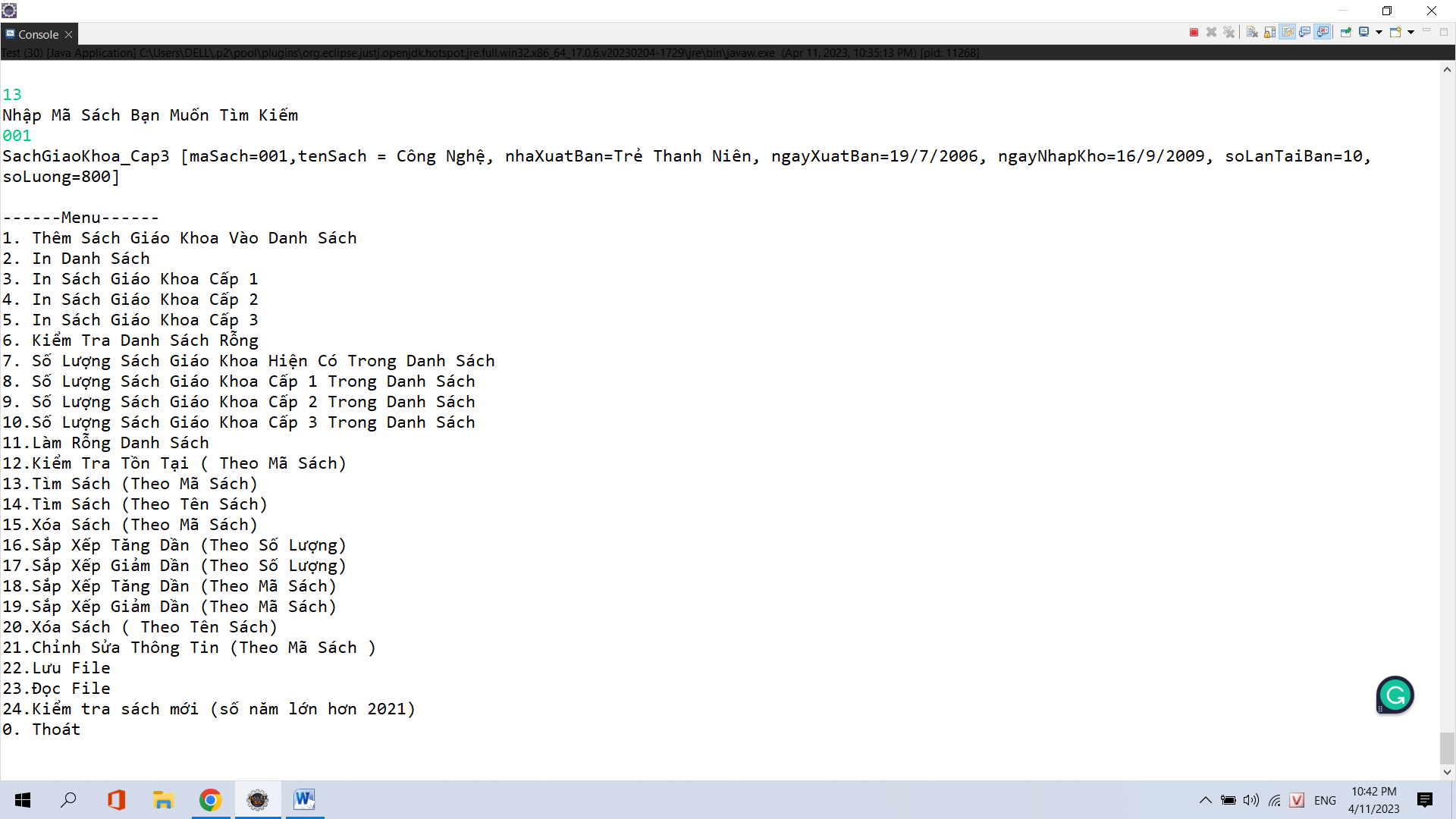




1. Kiểm tra sách có nằm trong kho không bằng mã sách



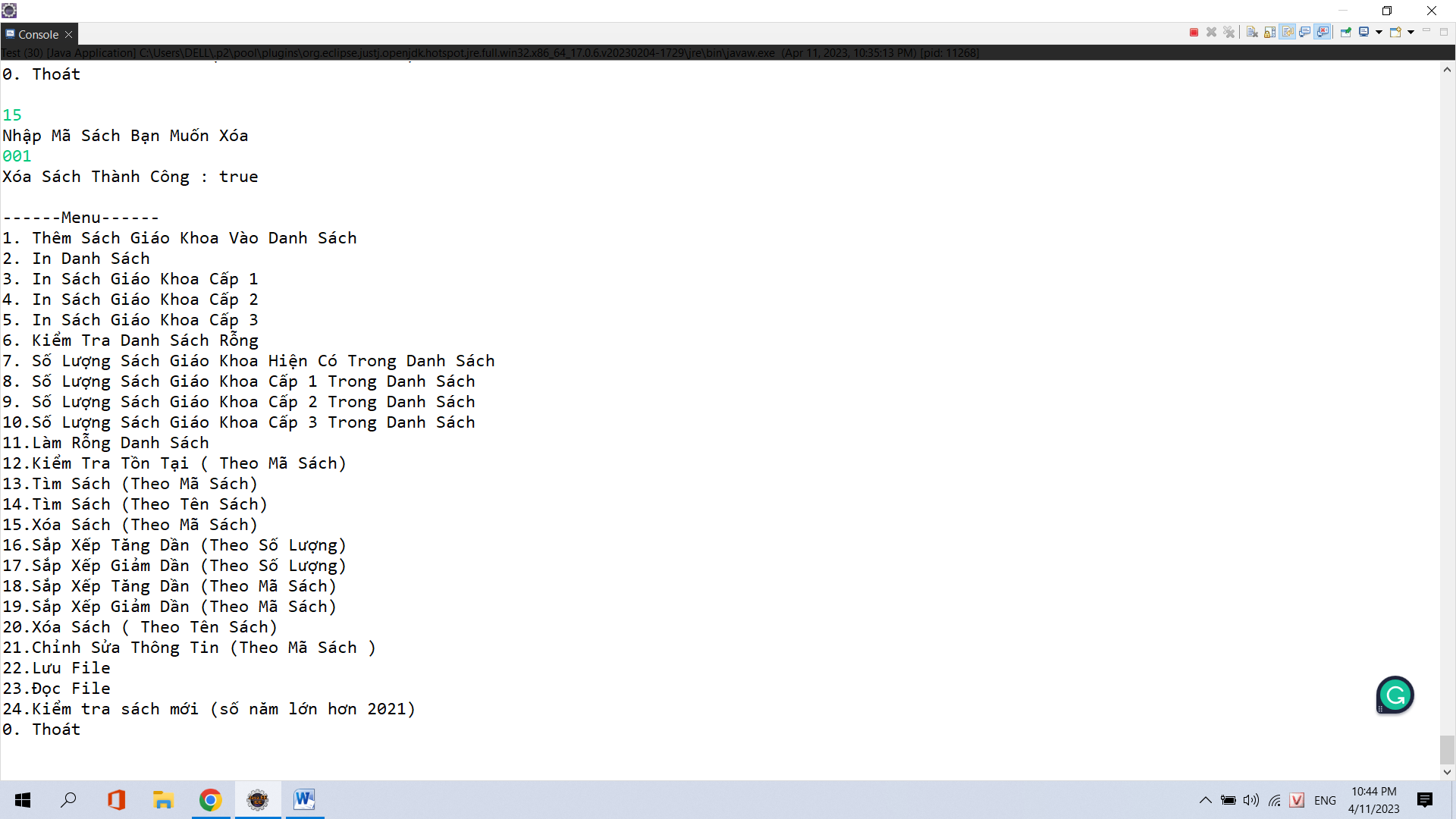
1. Tìm sách bằng mã sách

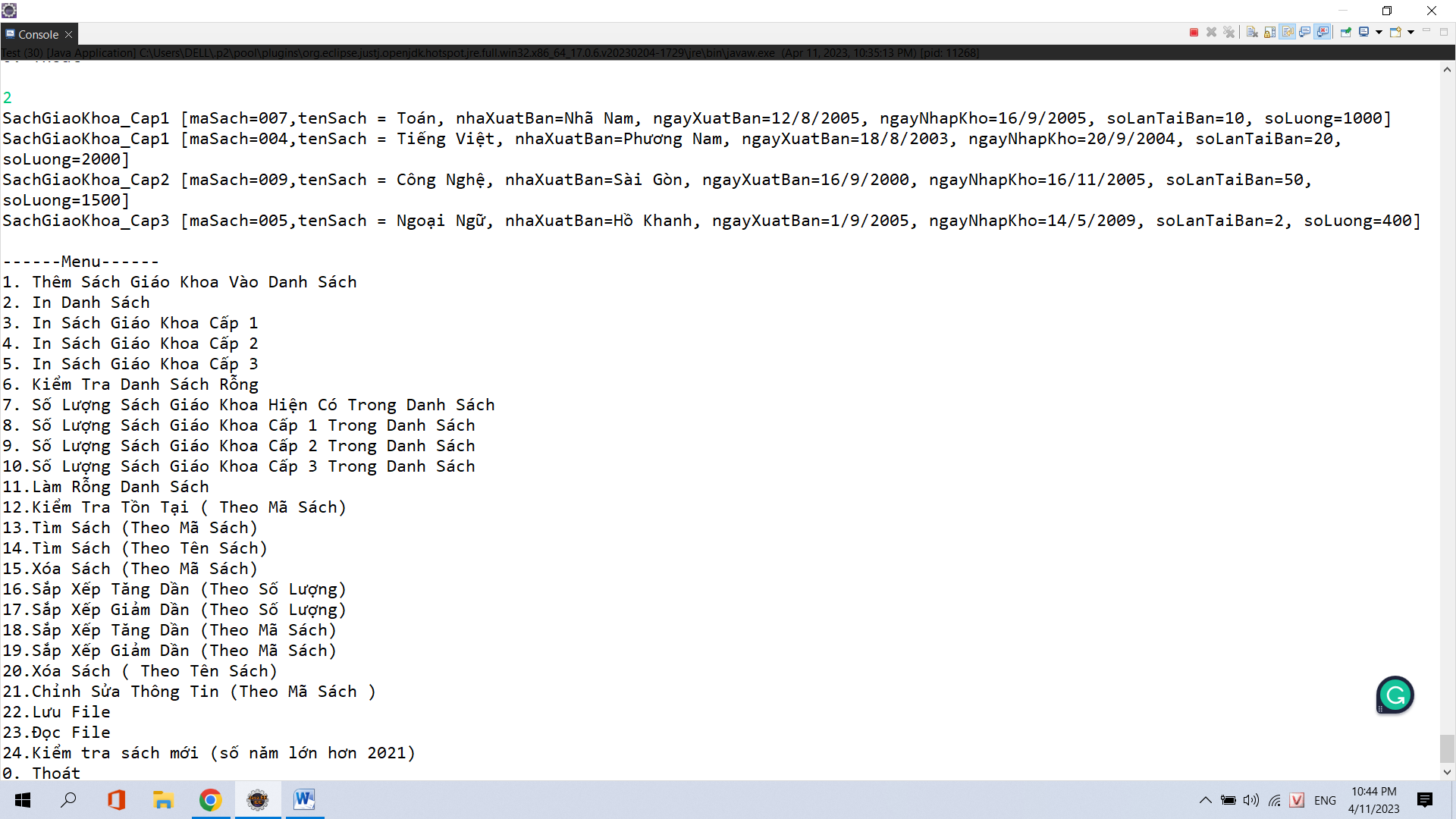


1. Tìm sách theo tên

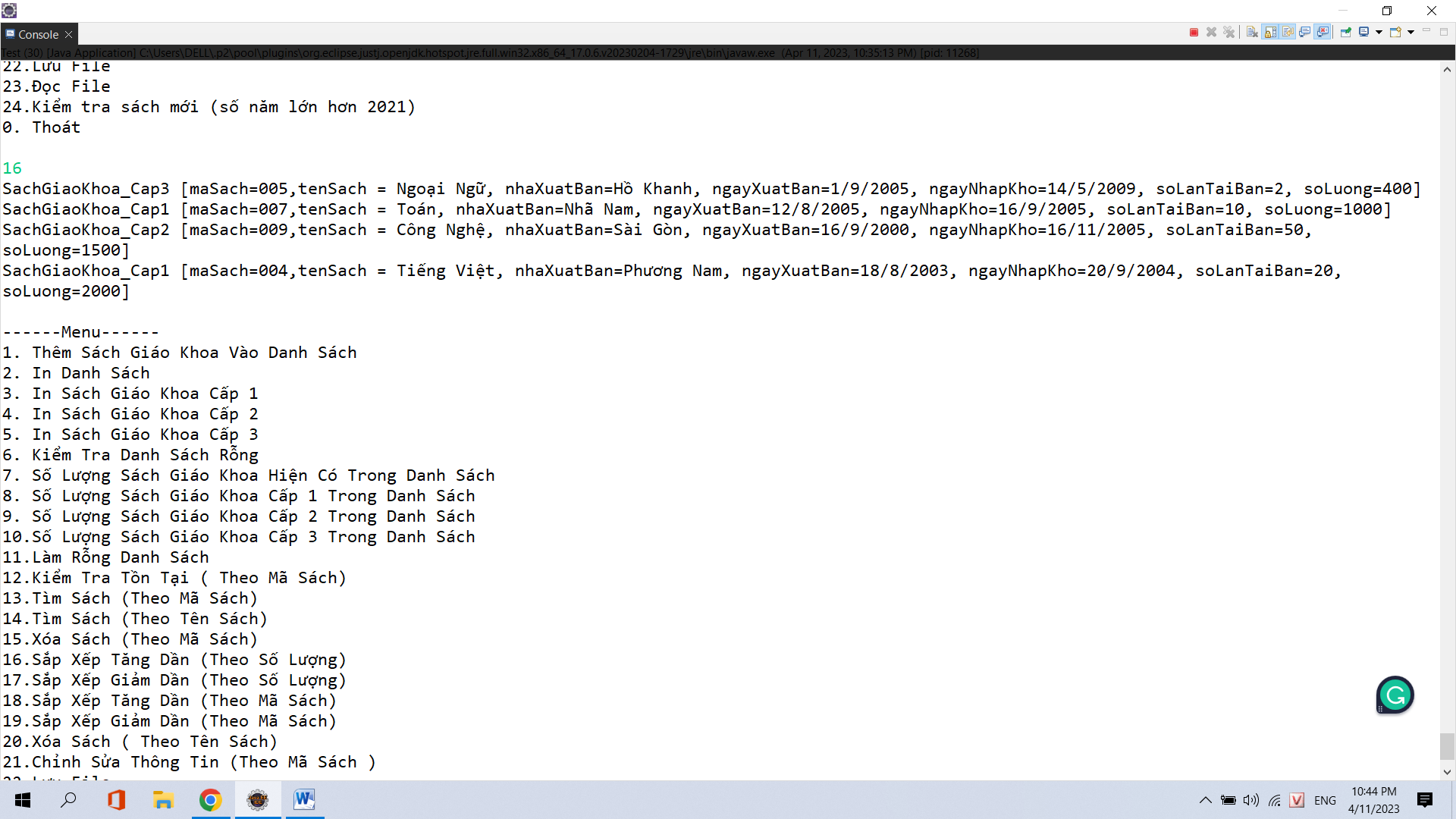


1. Xóa sách theo mã sách

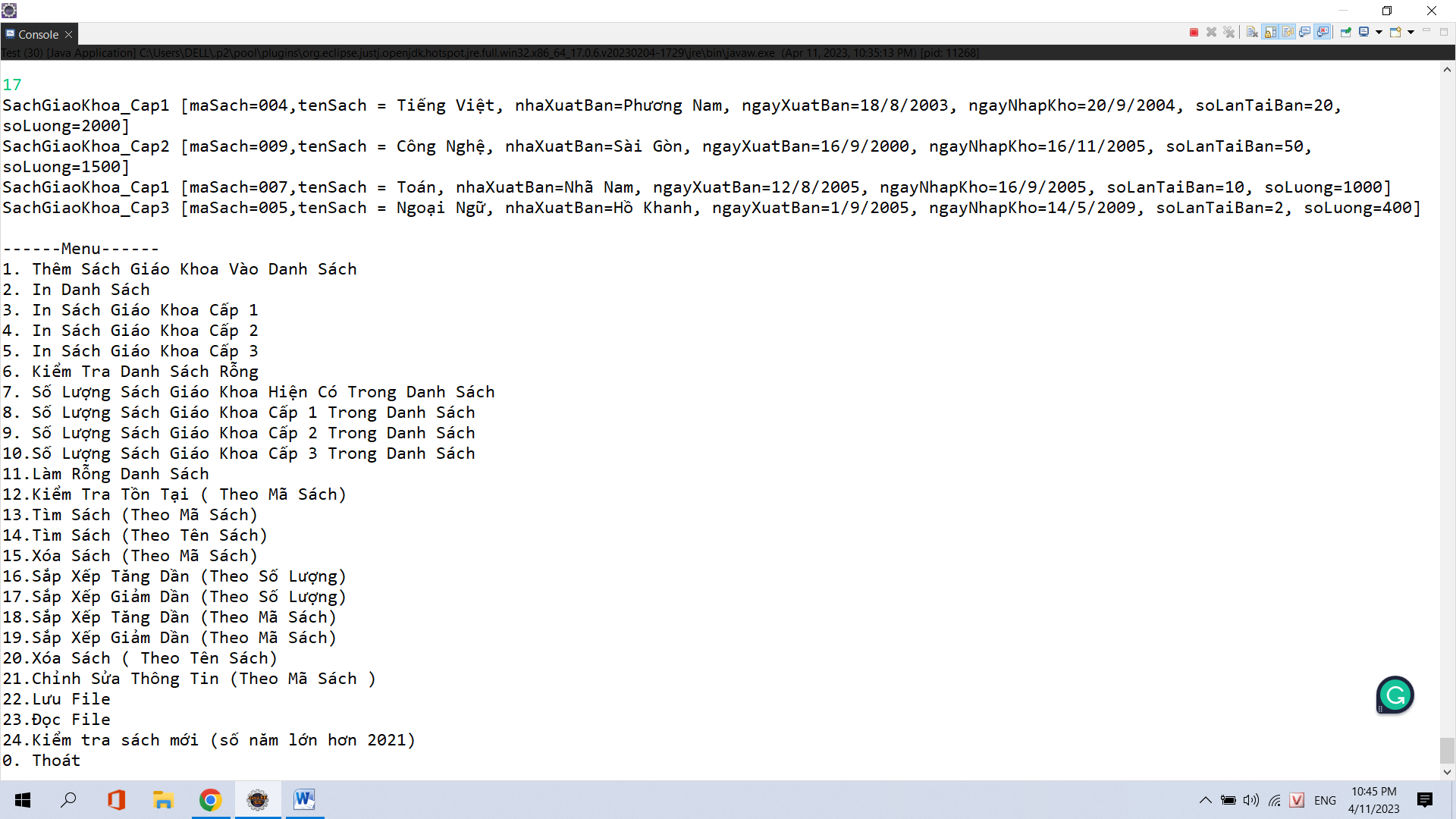




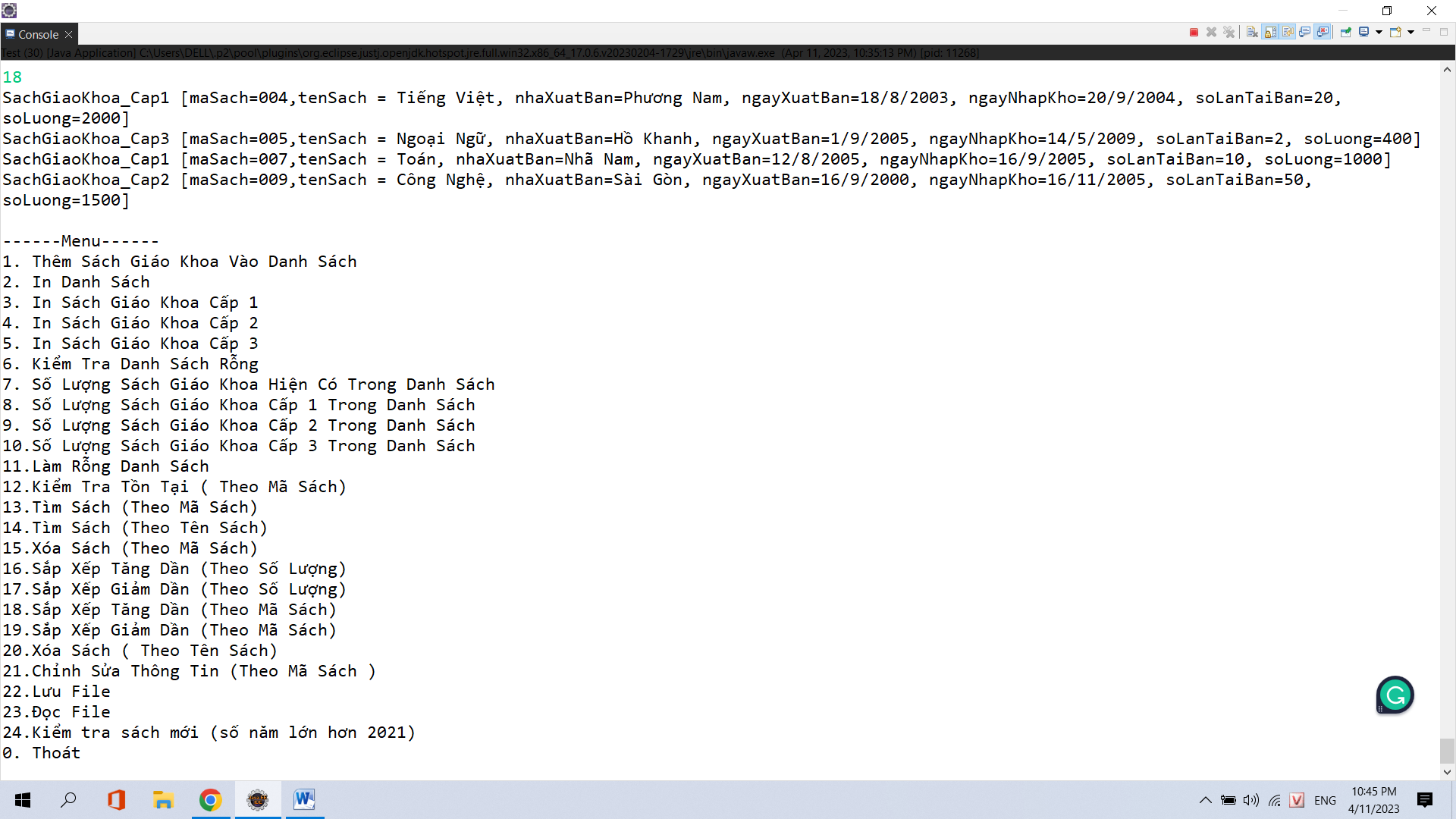
1. Xuất danh sách theo thư tứ tăng dần của số lượng sách



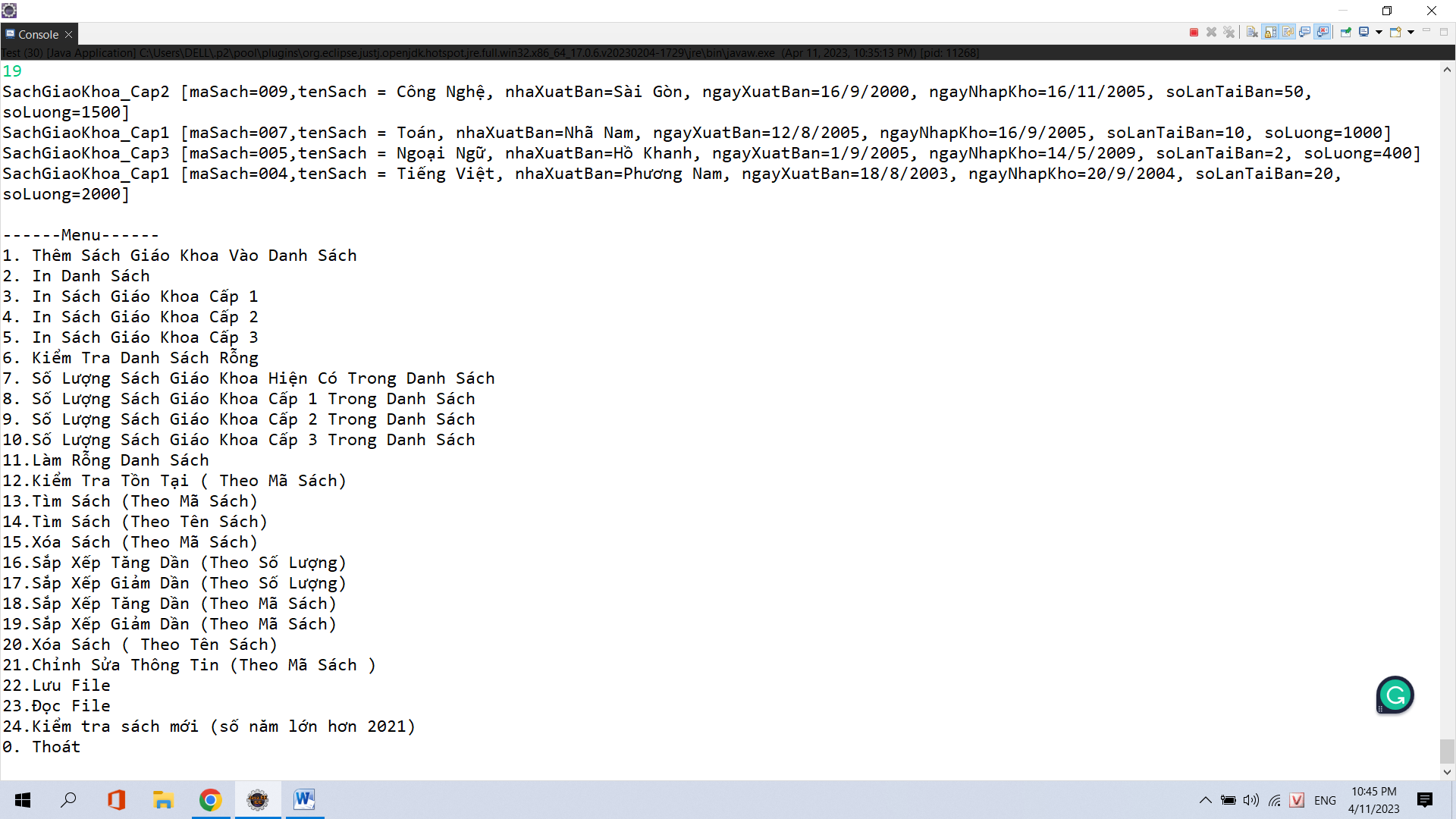
1. Xuất danh sách theo thứ tự giảm dần của số lượng sách



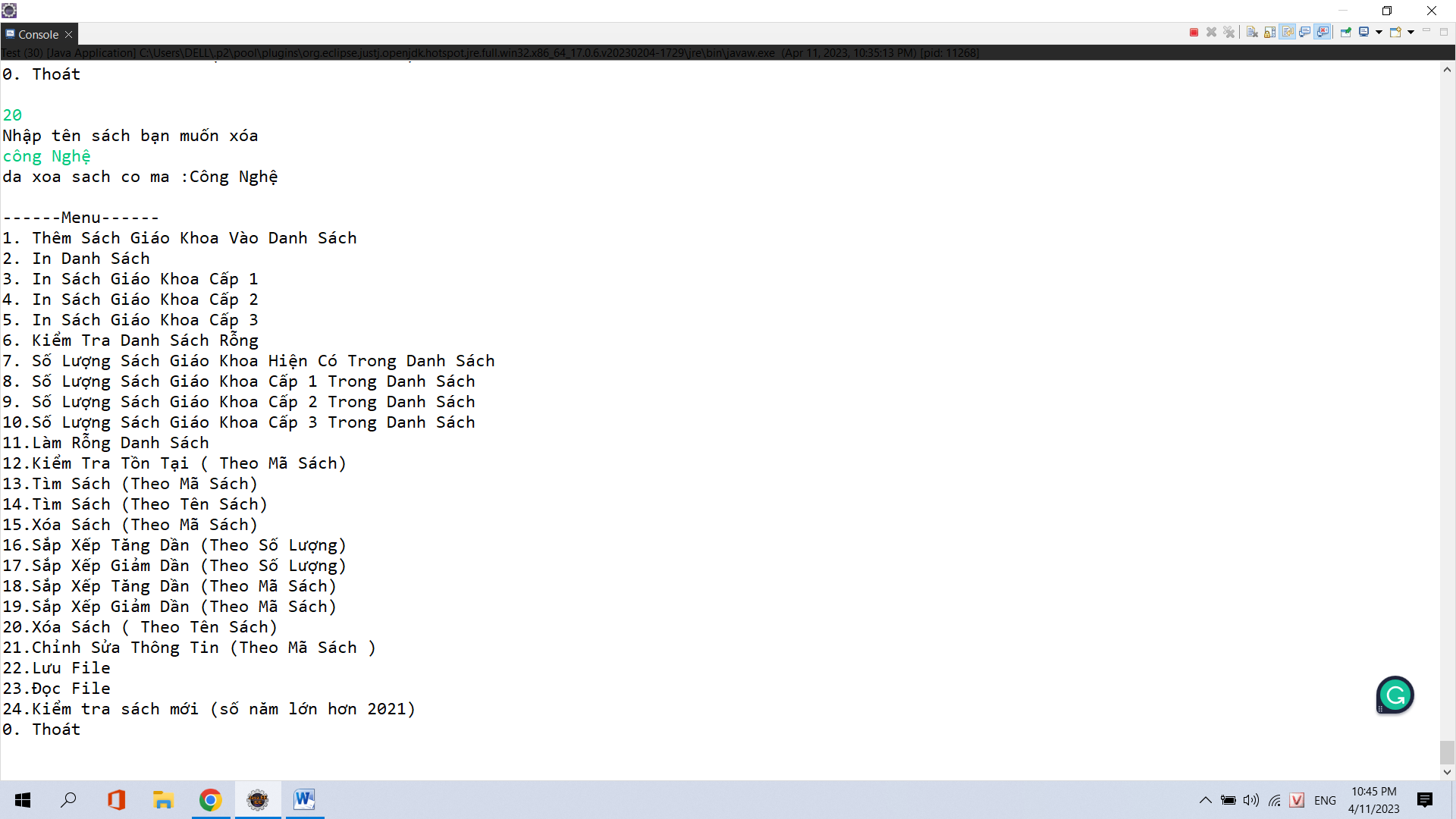
18.Xuất danh sách theo thứ tự tăng dần theo mã sách

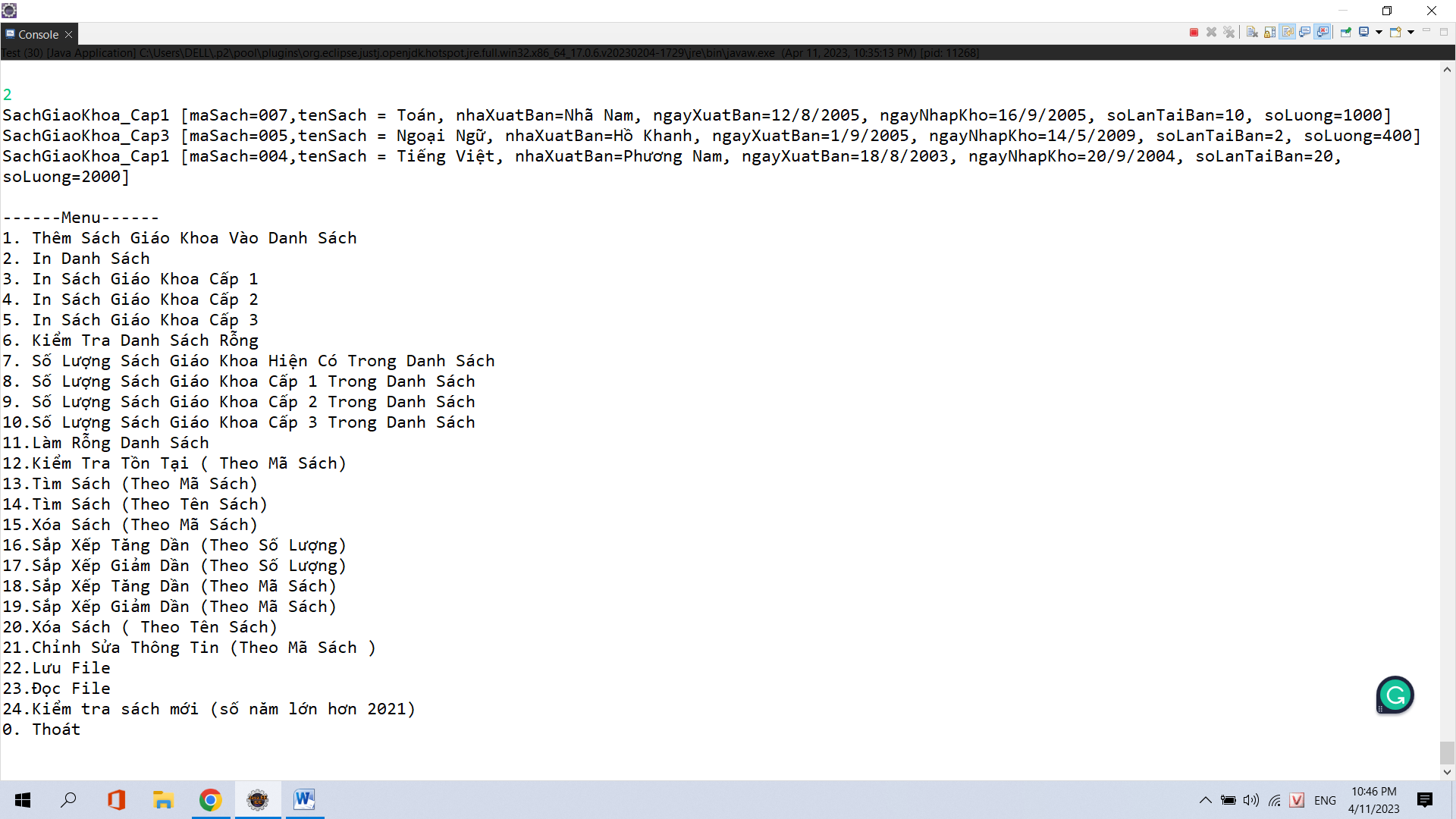


19.Xuất danh sach theo thứ tự giảm dần theo mã sách

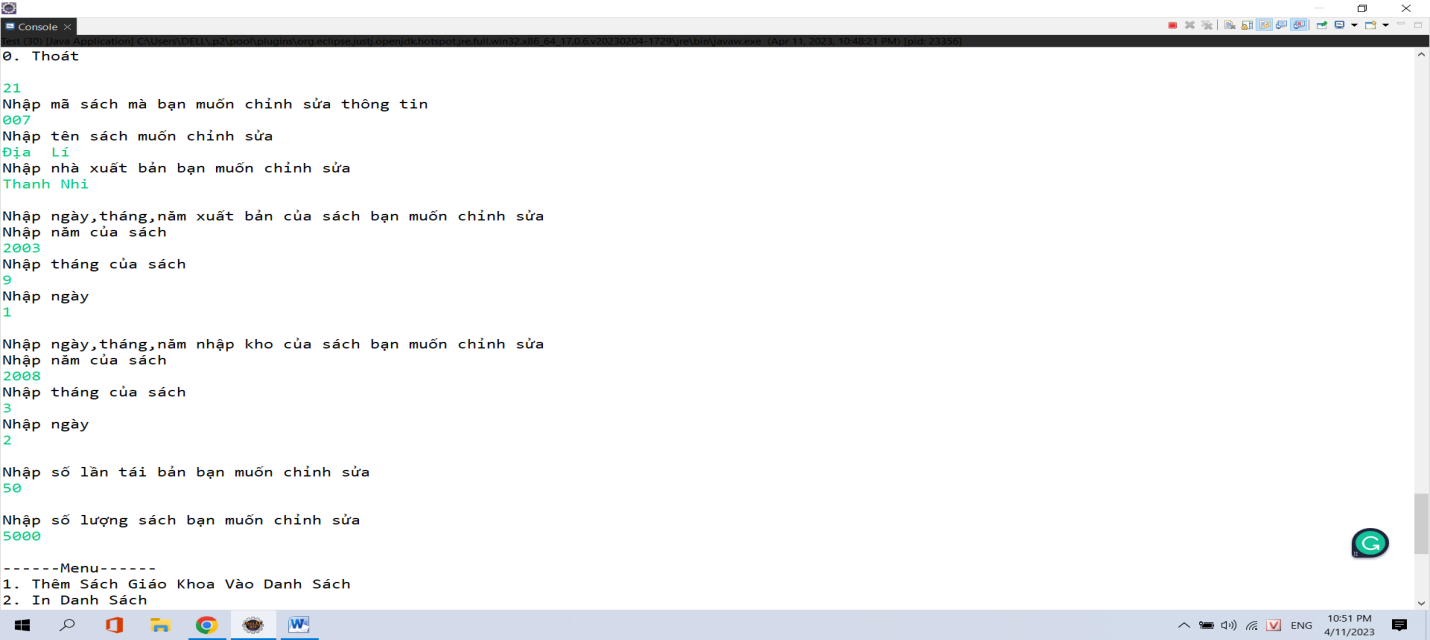


1. Xóa sách theo tên

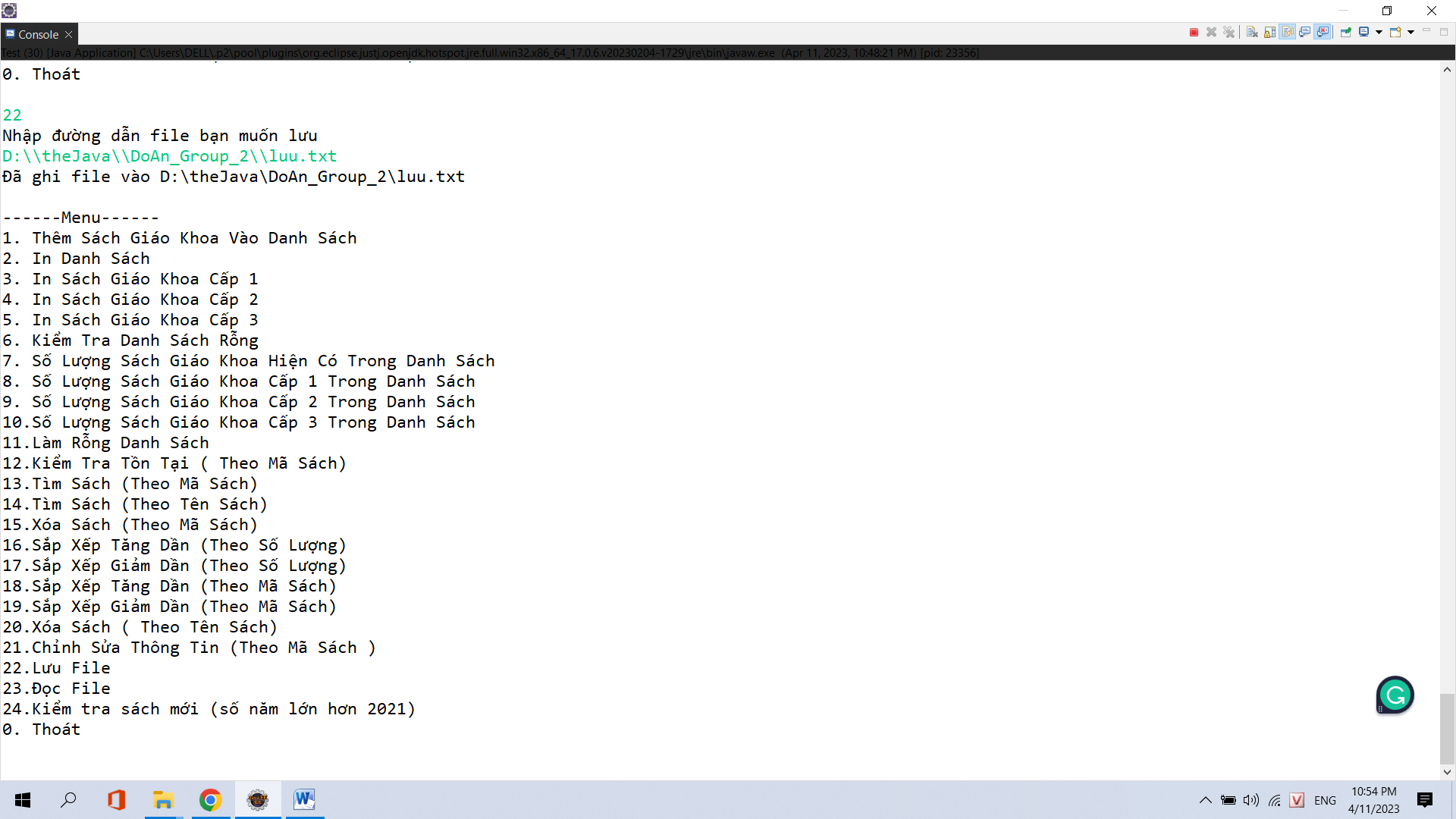




1. Chỉnh sửa thông tin



1. Lưu dữ liệu vào file

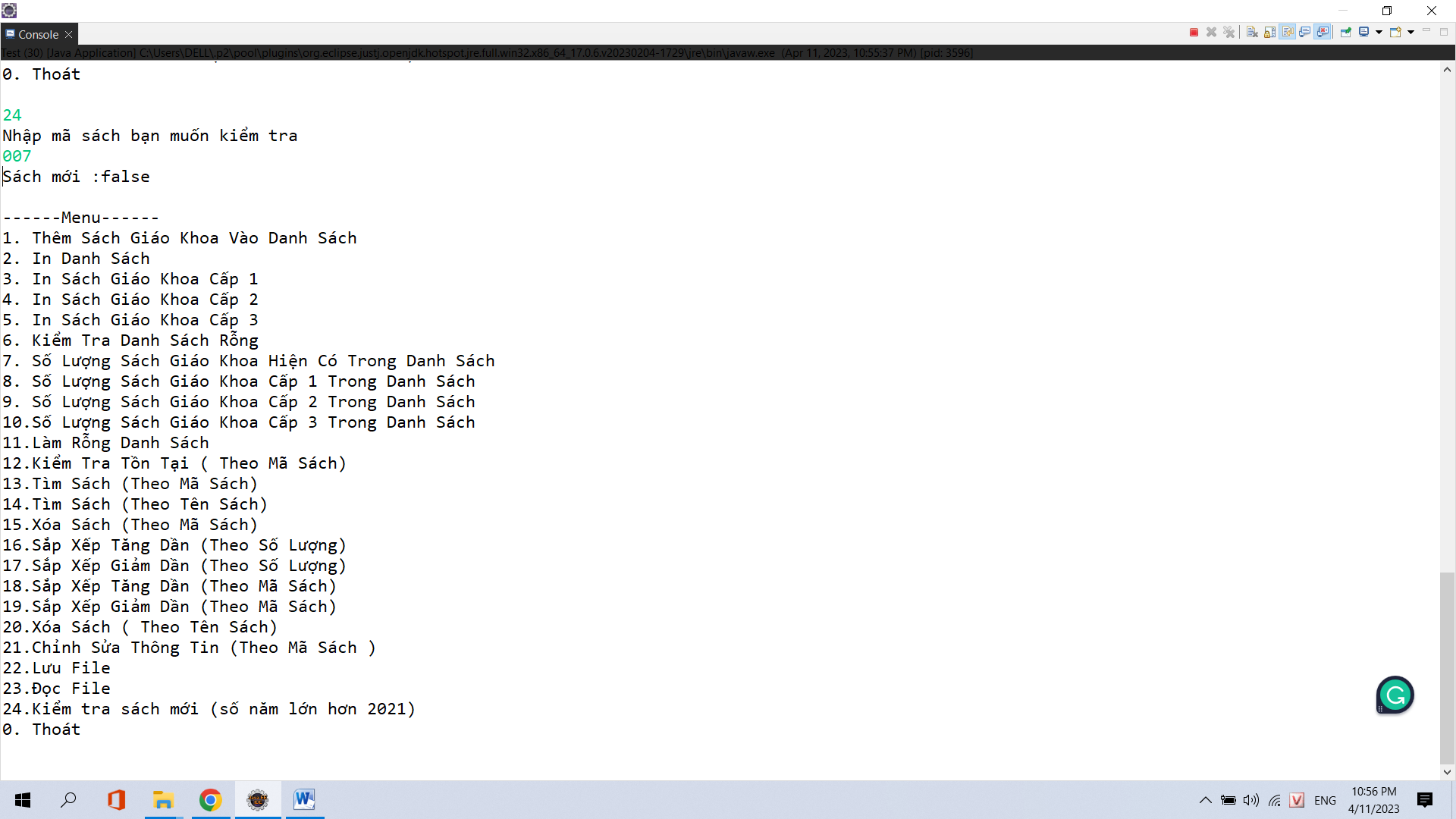


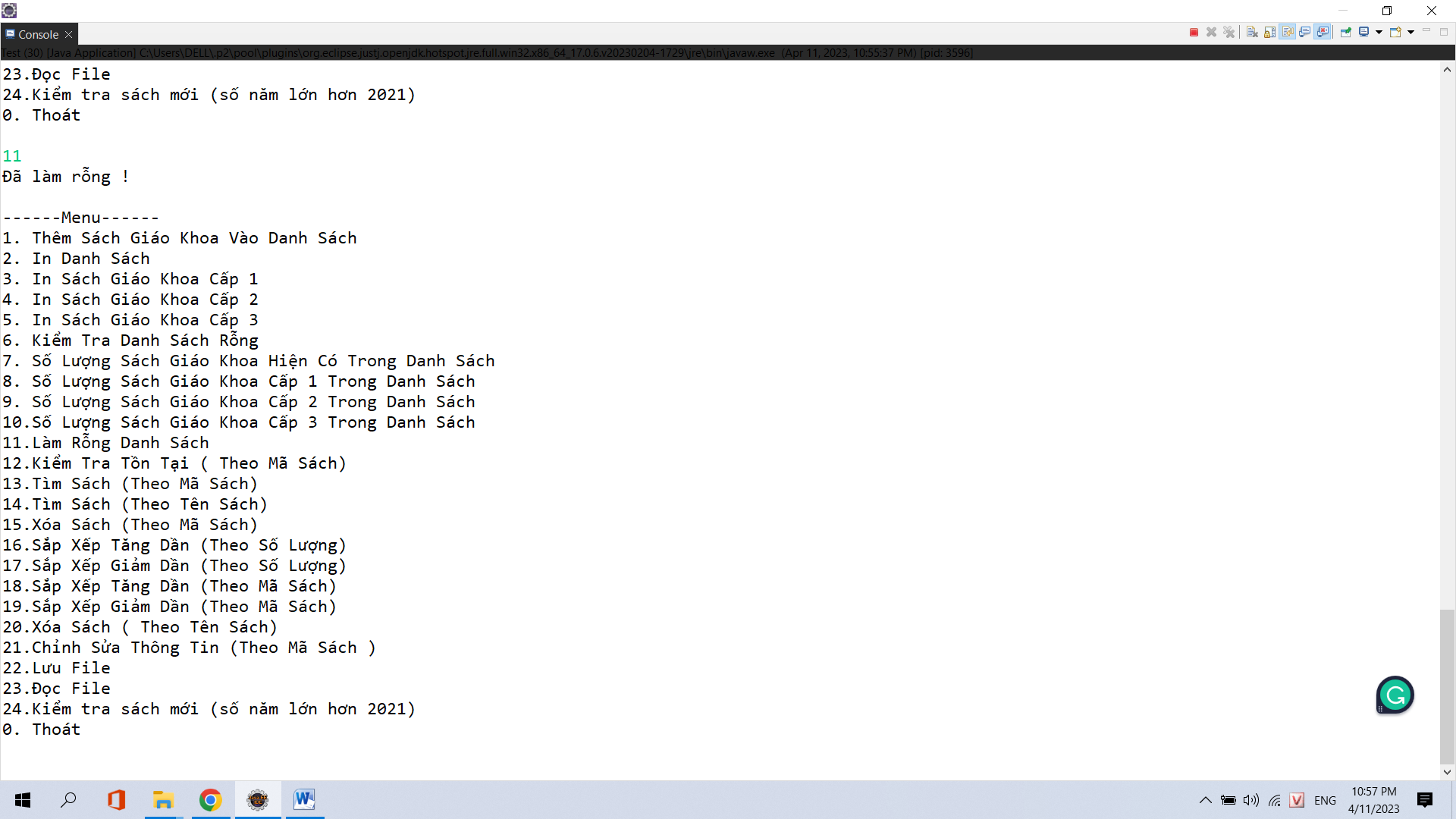


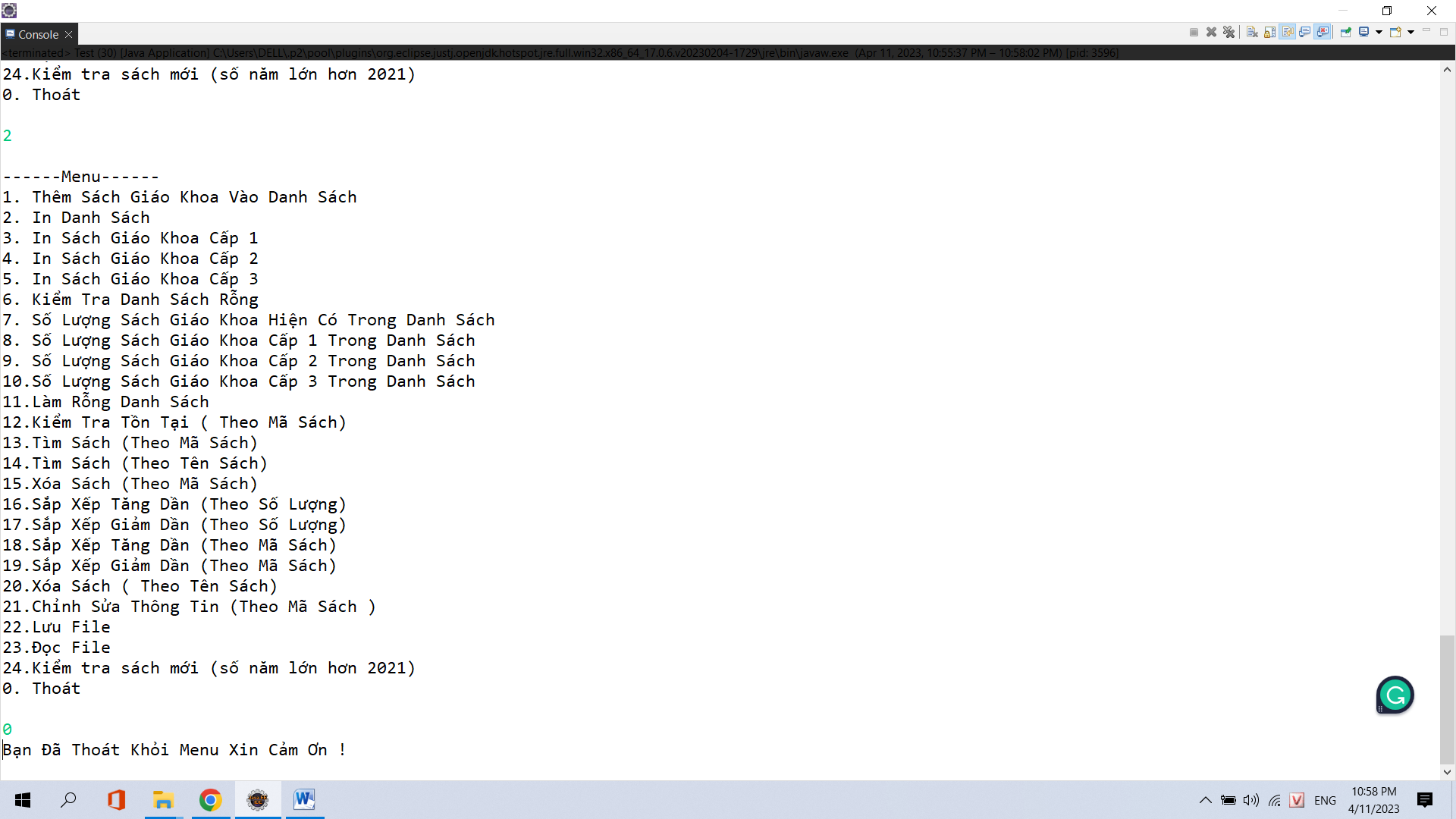
23.Đọc dữ liệu từ file



24.Kiểm tra sách mới







**Tài liệu tham khảo:**

[[Ebook] Tài liệu học lập trình Java từ cơ bản đến nâng cao - ĐH FPT Polytechnic (PDF) - Technology Diver (cuongquach.com)](https://cuongquach.com/laptrinh-java-coban-nangcao-fpt-polytechnic.html)

[Lập trình Java (Java Core) – TITV](https://titv.vn/courses-page/lap-trinh-java-java-core/)

[Bao Cao Mau Mon UML Quan Ly Kho Vat Tu - CHƢƠNG 1 TỔNG QUAN ĐỀ TÀI 1. TÊN ĐỀ TÀI “XÂY DỰNG PHẦN MỀM - Studocu](https://www.studocu.com/vn/document/truong-dai-hoc-sai-gon/he-thong-thong-tin/bao-cao-mau-mon-uml-quan-ly-kho-vat-tu/18150790)

[Báo cáo Tổng quan về lập trình hướng đối tượng - Tài liệu, ebook, giáo trình (doc.edu.vn)](https://doc.edu.vn/tai-lieu/bao-cao-tong-quan-ve-lap-trinh-huong-doi-tuong-55128/)