

**MỤC LỤC**

[**I. Phát biểu bài toán** 3](#_Toc132134372)

[**II. Xác định các lớp của bài toán** 3](#_Toc132134373)

[**1. Class sach** 3](#_Toc132134374)

[**2. Class sachhoctap kế thừa từ class sach** 3](#_Toc132134375)

[**3. class sachgiaitri kế thừa từ class sach** 3](#_Toc132134376)

[**4. class QuanLy** 4](#_Toc132134377)

[**5. class sapxep** 4](#_Toc132134378)

[**III. Mô tả thuật toán thao tác** 6](#_Toc132134379)

[**1. Chỉnh sửa thông tin sách học tập, sách giải trí** 6](#_Toc132134380)

[**2.Tìm kiếm sách** 6](#_Toc132134381)

[**3. Sắp xếp sách theo số lượng tăng dần** 6](#_Toc132134382)

[**4. Hiện ra thông tin các sách** 6](#_Toc132134383)

[**IV. Cài đặt bài toán** 6](#_Toc132134384)

[**1. Class sach** 6](#_Toc132134385)

[**2. Class sachhoctap** 9](#_Toc132134386)

[**3. Class sachgiaitri** 9](#_Toc132134387)

[**4. Class Sapxep** 10](#_Toc132134388)

[**5. Class QuanLy** 10](#_Toc132134389)

[**6. Class Main** 15](#_Toc132134390)

[**V. Kiểm thử lập bộ test** 16](#_Toc132134391)

[**1. Nhập thông tin sách học tập** 17](#_Toc132134392)

[**2. Nhập thông tin sách giải trí** 18](#_Toc132134393)

[**3. Hiện thông tin sách học tập** 18](#_Toc132134394)

[**4. Hiện thông tin sách giải trí** 18](#_Toc132134395)

[**5. Chỉnh sửa thông tin( Thêm, Sửa, Xóa)** 19](#_Toc132134396)

[**6. Sắp xếp danh sách sách theo số lượng tăng dần** 20](#_Toc132134397)

[**7. Hiển thị sách cần tìm** 21](#_Toc132134398)

# **I. Phát biểu bài toán**

Quản lý kho sách luôn là công việc cần thiết khi mà hằng ngày mặt hàng này nhập xuất kho liên tục. Vậy làm thế nào để giải bài toán quản lý kho sách. Đây là đề tài của nhóm chúng em, sẽ giải đáp trong những phần dưới đây.

Các đối tượng cần quản lý:

**Sách giải trí**: mỗi sách giải trí gồm có Mã sách(duy nhất), Tên sách , nhà xuất bản, đơn giá, số lượng.

**Sách học tập** : mỗi sách học tập gồm có Mã sách(duy nhất), Tên sách , nhà xuất bản, đơn giá, số lượng.

**Sách giải trí** và **Sách học tập** cùng có các thuộc tính như Mã sách(duy nhất), Tên sách , nhà xuất bản, đơn giá, số lượng kế thừ từ lớp sách

* Tính kế thừa

ArrayList<sach> quản lý cả Sách học tập, Sách giải trí, Sách, khi nhập ta có thể nhập tùy theo Đối tượng Sách học tập hoặc Sách vào ArrayList này

* Tính đa hình

# **II. Xác định các lớp của bài toán**

## **1. Class sach**

* Thuộc tính: String maSach, tensach, nhaXuatBan, double donGia, int soLuong.
* Phương thức:
  + Constructor: Hàm tạo có đối số và hàm tạo không đối số
  + Getter, Setter: Phương thức get, set cho các thuộc tính kiểu public
  + Nhap: Nhập thông tin các thuộc tính
  + Hien: Hiện ra các thông tin

## **2. Class sachhoctap kế thừa từ class sach**

* Thuộc tính:
* Phương thức:
  + Hien: override hien() ở class sach, hiện thông tin sách học tập
  + hiensachhoctap: chỉ hiện thông tin Sachhoctap

## **3. class sachgiaitri kế thừa từ class sach**

* Thuộc tính:
* Phương thức:
  + Hien: override phương thức hien() ở class sach, hiện thị thông tin sách giải trí

## **4. class QuanLy**

* Thuộc tính:
  + ArrayList<sach> s
* Phương thức:
  + themsach(): thêm Sách, có thể là Sách học tập hoặc Sách giải trí vào ArrayList<sach>
  + menuSuaDoi(): Menu hiện thị các chức năng sửa đổi ở SuaDoi()
  + SuaDoi(): Thêm, Sửa các thông tin như Sách học tập, Sách giải trí
  + suasachhoctap(): sửa thông tin sách học tập
  + xoasachhoctap(): xóa sách học tập theo mã sách
  + suasachgiaitri(): sửa thông tin sách giải trí
  + xoasachgiaitri(): xóa sách giải trí theo mã sách
  + xuatsht(): hiển thị danh sách sách học tập
  + xuatsgt(): hiện thông tin sách giải trí
  + SapXep(): Sắp xếp sách theo thứ tự số lượng tăng dần
  + menuChinh(): menu chương trình chính
  + timkiem(): tìm kiếm sách theo tên sách

## **5. class sapxep**

Comparator<sach>: để sắp xếp các đối tượng của lớp sach

SapXep(): Collections Sort

|  |
| --- |
| **sach** |
| -String maSach  -String tenSach  -String nhaXuatBan  -double donGia  -int soLuong  +Constructer()  +Setter(),Getter()  +nhap()  +hien() |

|  |
| --- |
| **sachhoctap** |
| + Constructer  + Getter,Setter  + nhap()  + hien() |

|  |
| --- |
| **sachgiaitri** |
| + Constructer  + Getter,Setter  + nhap()  + hien() |

|  |
| --- |
| **QuanLy** |
| - ArrayList<sach> s  + Constructor  + themsach()  + menuSuaDoi()  + SuaDoi()  + suasachhoctap()  + xoasachhoctap()  + suasachgiaitri()  + xoasachgiaitri()  + xuatsht()  + xuatsgt()  + menuChinh()  + timkiem() |

|  |
| --- |
| **sapxep** |
| +Comperator<sach>  +Sapxep():Collection Sort |

|  |
| --- |
| **Main** |
| * **main()** |

# **III. Mô tả thuật toán thao tác**

## **1. Chỉnh sửa thông tin sách học tập, sách giải trí**

Thuật toán:

Nhập vào maSach cần tìm để chỉnh sửa

Dùng vòng lặp và câu lệnh so sánh, nếu từ khóa truyền vào trùng với dữ liệu đã có sẽ hiển thị ra để thao tác chỉnh sửa, không có sẽ không thông báo gì

Xóa: Trong ArrayList dùng phép remove

Sửa : Nhập lại thông tin mới

## **2.Tìm kiếm sách**

Thuật toán:

Trong ArrayList đã nhập, tìm kiếm và hiện ra những sách có mã sách trùng với thông tin nhập vào.

## **3. Sắp xếp sách theo số lượng tăng dần**

Thuật toán:

Sử dụng Collections Sort trong ArrayList<sach>

Tạo class SapXep, với phương thức int compare(

sach o1, sach o2)

Đặt sx= o1.getSoLuong()-o2.getSoLuong();

Nếu sx < 0 return -1

Nếu sx > 0 return 1

Nếu sx =0 return 0

## **4. Hiện ra thông tin các sách**

Thuật toán:

Trong ArrayList<sach>, lọc ra Sách là Sách giải trí hay sách học tập bằng phép instance of

# **IV. Cài đặt bài toán**

## **1. Class sach**

package qlks;

import java.util.Scanner;

public class sach {

    String maSach;

    String tensach;

    String nhaXuatBan;

    double donGia;

    int soLuong;

    public sach() {

        super();

    }

    public sach(String maSach,String tensach, String nhaXuatBan, double donGia, int soLuong ) {

        super();

        this.maSach = maSach;

        this.tensach=tensach;

        this.nhaXuatBan = nhaXuatBan;

        this.donGia = donGia;

        this.soLuong = soLuong;

    }

    public String getMaSach() {

        return this.maSach;

    }

    public void setMaSach(String maSach) {

        this.maSach = maSach;

    }

    public String getTensach() {

        return this.tensach;

    }

    public void setTensach(String tensach) {

        this.tensach=tensach;

    }

    public String getNhaXuatBan() {

        return this.nhaXuatBan;

    }

    public void setNhaXuatBan(String nhaXuatBan) {

        this.nhaXuatBan = nhaXuatBan;

    }

    public double getDonGia() {

        return this.donGia;

    }

    public void setDonGia(double donGia) {

        this.donGia = donGia;

    }

    public int getSoLuong() {

        return this.soLuong;

    }

    public void setSoLuong(int soLuong) {

        this.soLuong = soLuong;

    }

    public void nhap() {

Scanner scanner = new Scanner(System.in);

        System.out.println("Nhập mã sách: ");

        maSach = scanner.nextLine();

        System.out.println("Nhập tên sách: ");

        tensach = scanner.nextLine();

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

        nhaXuatBan = scanner.nextLine();

        System.out.println("Nhập đơn giá: ");

        donGia = scanner.nextDouble();

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

        soLuong = scanner.nextInt();

    }

    public void hien()

{

System.out.print("{mã sách: "+ getMaSach() + ", tên sách: "+getTensach() +", nhà xb: "+ getNhaXuatBan()+ ", đơn giá: "+getDonGia() +"00 "+ "số lượng: "+getSoLuong()+"}");

}

}

## **2. Class sachhoctap**

package qlks;

public class sachhoctap extends sach{

public sachhoctap() {

}

public sachhoctap(String maSach,String tensach, String nhaXuatBan, double donGia, int soLuong)

{

super( maSach, tensach, nhaXuatBan,donGia,soLuong);

}

@Override

public void hien()

{

System.out.print("{mã sách: "+ getMaSach() +", tên sách: "+getTensach() +", nhà xb: "+ getNhaXuatBan()+ ", đơn giá: "+getDonGia() +"00 "+ "số lượng: "+getSoLuong()+"}");

System.out.print("loại sách học tập");

super.hien();

}

}

## **3. Class sachgiaitri**

package qlks;

public class sachgiaitri extends sach{

public sachgiaitri() {

}

public sachgiaitri(String maSach,String tensach, String nhaXuatBan, double donGia, int soLuong) {

super( maSach, tensach, nhaXuatBan,donGia,soLuong);

}

@Override

public void hien()

{

System.out.print("{mã sách: "+ getMaSach() +", tên sách: "+getTensach() +", nhà xb: "+ getNhaXuatBan()+ ", đơn giá: "+getDonGia() +"00 "+ "số lượng: "+getSoLuong()+"}");

System.out.print("loại sách giải trí");

super.hien();

}

}

## **4. Class Sapxep**

package qlks;

import java.util.Comparator;

//Comparator: để sắp xếp các đối tượng của lớp do người dùng định nghĩa

public class sapxep implements Comparator<sach>{

    @Override

    public int compare(sach o1, sach o2) {

        int sx=o1.getSoLuong()-o2.getSoLuong();

        if(sx< 0)

return -1;

else if(sx ==0)

return 0;

else

return 1;

    }

}

## **5. Class QuanLy**

package qlks;

import java.util.ArrayList;

import java.util.Collections;

import java.util.Scanner;

public class QuanLy {

ArrayList<sach> s = new ArrayList<>();;

public QuanLy() {

}

public void themsach(sach a) {

Scanner sc = new Scanner(System.in);

System.out.print("Nhập số lượng cần thêm: ");

int n = sc.nextInt();

for (int i = 0; i < n; i++) {

System.out.println("Lần nhập thứ " + (i + 1) + ": ");

if (a instanceof sachhoctap) {

a = new sachhoctap();

a.nhap();

} else if (a instanceof sachgiaitri) {

a = new sachgiaitri();

a.nhap();

}

s.add(a);

}

}

//instanceof trường hợp của

public void xuatsgt() {

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

for (sach x : s) {

    if (x instanceof sachgiaitri) {

x.hien();

}

}

}

public void xuatsht() {

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

for (sach x : s) {

if (x instanceof sachhoctap) {

x.hien();

}

}

}

public void SuaDoi() {

Scanner sc = new Scanner(System.in);

int n;

do {

    System.out.println("Chọn: ");

    System.out.println("1. Sửa sách học tập");

    System.out.println("2. Xóa sách học tập");

    System.out.println("3. Sửa sách giải trí");

    System.out.println("4. Xóa sách giải trí");

    System.out.println("0. Quay lại");

    n = sc.nextInt();

switch (n) {

case 1: {

System.out.print("Nhập mã sách học tập cần sửa: ");

sc.nextLine();

String maSachht = sc.nextLine();

suasachhoctap(maSachht);

break;

}

case 2: {

System.out.print("Nhập mã sách học tập cần xóa: ");

sc.nextLine();

String maSachht = sc.nextLine();

xoasachhoctap(maSachht);

break;

}

case 3: {

System.out.print("Nhập mã sách giải trí sửa thông tin: ");

sc.nextLine();

String maSachgt = sc.nextLine();

suasachgiaitri(maSachgt);

break;

}

case 4: {

System.out.print("Nhập mã sách giải trí cần xóa : ");

sc.nextLine();

String maSachgt = sc.nextLine();

xoasachgiaitri(maSachgt);

break;

}

default:

break;

}

} while (n != 0);

}

//compareTo(: so sách với

public void suasachhoctap(String maSachht) {

for (sach x : s) {

if (x instanceof sachhoctap) {

if (((sachhoctap) x).getMaSach().compareTo(maSachht) == 0) {

    x.nhap();

}

}

}

}

public void xoasachhoctap(String maSachht) {

for (sach x : s) {

if (x instanceof sachhoctap) {

if (((sachhoctap) x).getMaSach().compareTo(maSachht) == 0) {

   s.remove(x);

}

}

}

}

public void suasachgiaitri(String maSachgt) {

for (sach x : s) {

if (x instanceof sachgiaitri) {

if (((sachgiaitri) x).getMaSach().compareTo(maSachgt) == 0) {

    x.nhap();

}

}

}

}

public void xoasachgiaitri(String maSachgt) {

for (sach x : s) {

if (x instanceof sachgiaitri) {

if (((sachgiaitri) x).getMaSach().compareTo(maSachgt) == 0) {

    s.remove(x);

}

}

}

}

public void timkiem(){

    Scanner sc = new Scanner(System.in);

    System.out.println("nhập tên sách cần tìm");

    String a;

    a = sc.nextLine();

    for (sach x : s) {

if (((sach) x).getTensach().compareTo(a) == 0){

    if(x instanceof sachhoctap) {

    System.out.println("sách học tập");

    x.hien();

    }

    else if (x instanceof sachgiaitri){

    System.out.println("sách giải trí");

    x.hien();

    }

}

    else{

        }

}

}

public void hienDS()

{

for(sach x: s)

{

x.hien();

}

}

public void SapXep()

{

Collections.sort(s, new sapxep());

System.out.println(" sach sau khi sap xep");

hienDS();

}

public void menuChinh() {

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

    System.out.println("1. Nhập sách học tập ");

    System.out.println("2. Nhập sách giải trí");

    System.out.println("3. xuất sách học tập");

    System.out.println("4. xuất sách giải trí");

    System.out.println("5. Sửa xóa sách");

    System.out.println("6. Tìm kiếm sách");

    System.out.println("7. Sắp xếp sách theo số lượng tăng dần");

    System.out.println("-----Nhấn phím 0 để thoát chương trình, xin cảm ơn!-----");

}

}

## **6. Class Main**

package qlks;

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

QuanLy a= new QuanLy();

sach b;

int n;

Scanner sc= new Scanner(System.in);

do

{

a.menuChinh();

System.out.print("Lựa chọn: ");

n= sc.nextInt();

switch(n)

{

case 1:

{

b= new sachhoctap();

a.themsach(b);

break;

}

case 2:

{

b= new sachgiaitri();

a.themsach(b);

break;

}

case 3:

{

a.xuatsht();

break;

}

case 4:

{

a.xuatsgt();

break;

}

case 5:

{

    a.SuaDoi();

    break;

}

case 6:

{

    a.timkiem();

    break;

}

case 7:

{

    a.SapXep();

    break;

}

case 8:

default:

break;

}

}

while(n!=0);

}

}

# **V. Kiểm thử lập bộ test**

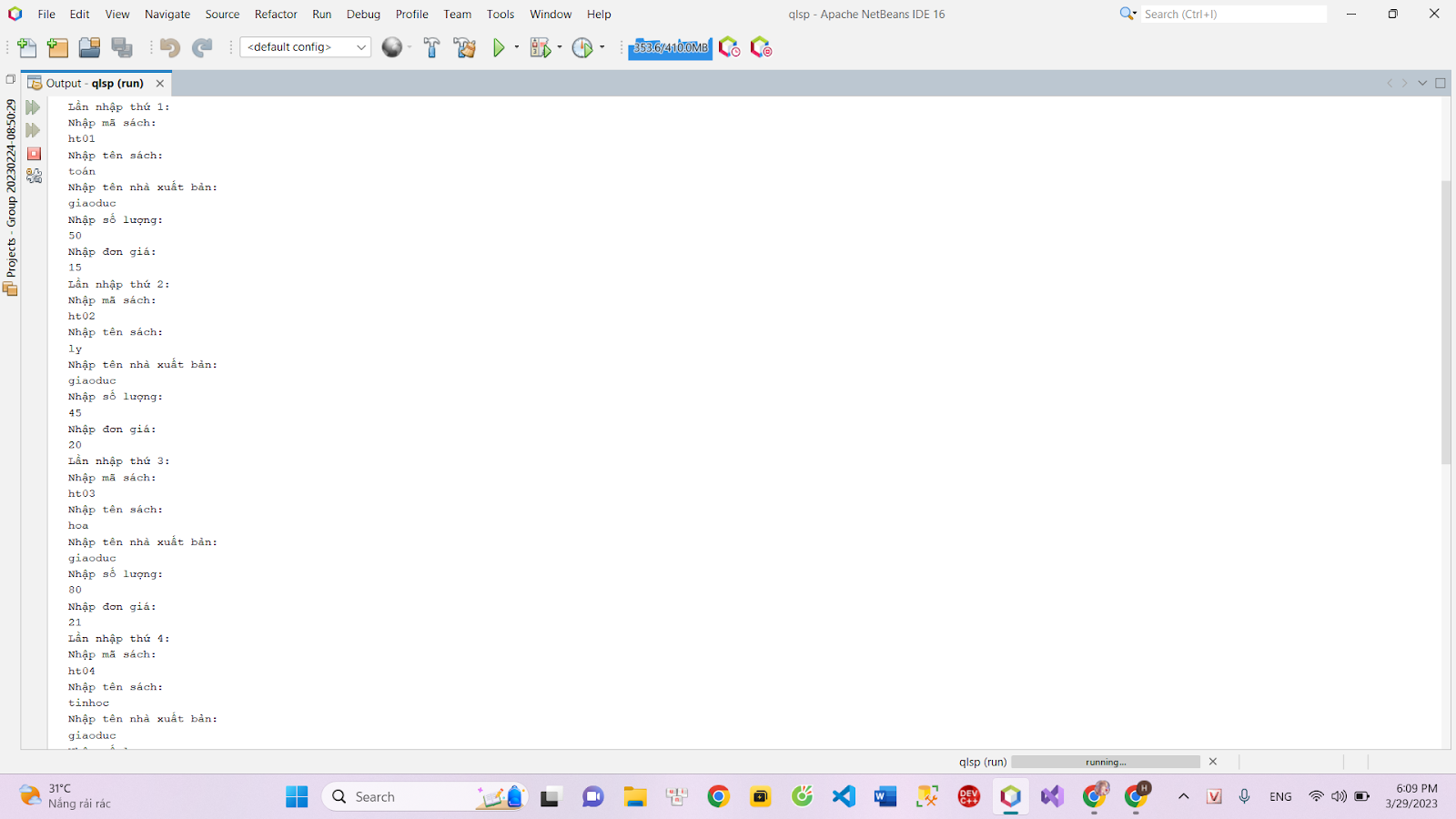
Dữ liệu sách học tập:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| MÃ SÁCH | TÊN SÁCH | NHÀ XUẤT BẢN | ĐƠN GIÁ | SỐ LƯỢNG |
| HT01 | Toán | nxb giáo dục | 15.000đ | 50 |
| HT02 | Lý | nxb giáo dục | 20.000đ | 45 |
| HT03 | Hóa | nxb giáo dục | 21.000đ | 80 |
| HT04 | Tin học | nxb giáo dục | 23.000đ | 30 |
| HT05 | Anh văn | nxb giáo dục | 18.000đ | 60 |

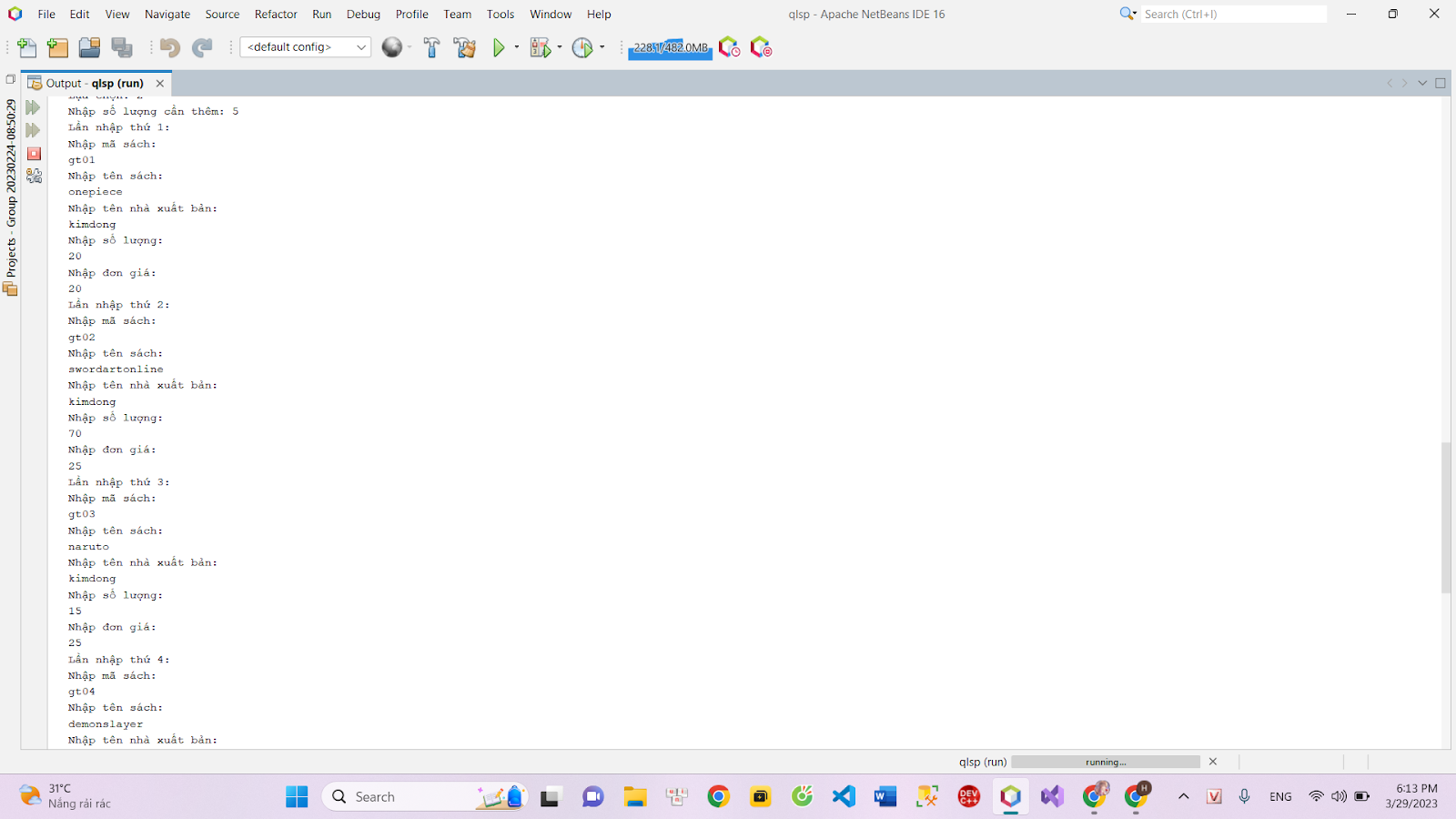
Dữ liệu sách giải trí:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| MÃ SÁCH | TÊN SÁCH | NHÀ XUẤT BẢN | ĐƠN GIÁ | SỐ LƯỢNG |
| GT01 | Onepiece | kim đồng | 20.000đ | 20 |
| GT02 | Sword Art Online ss1 | kim đồng | 25.000đ | 70 |
| GT03 | Naruto | kim đồng | 20.000đ | 15 |
| GT04 | Demon slayer | kim đồng | 20.000đ | 56 |
| GT05 | Date a live | kim đồng | 20.000đ | 81 |

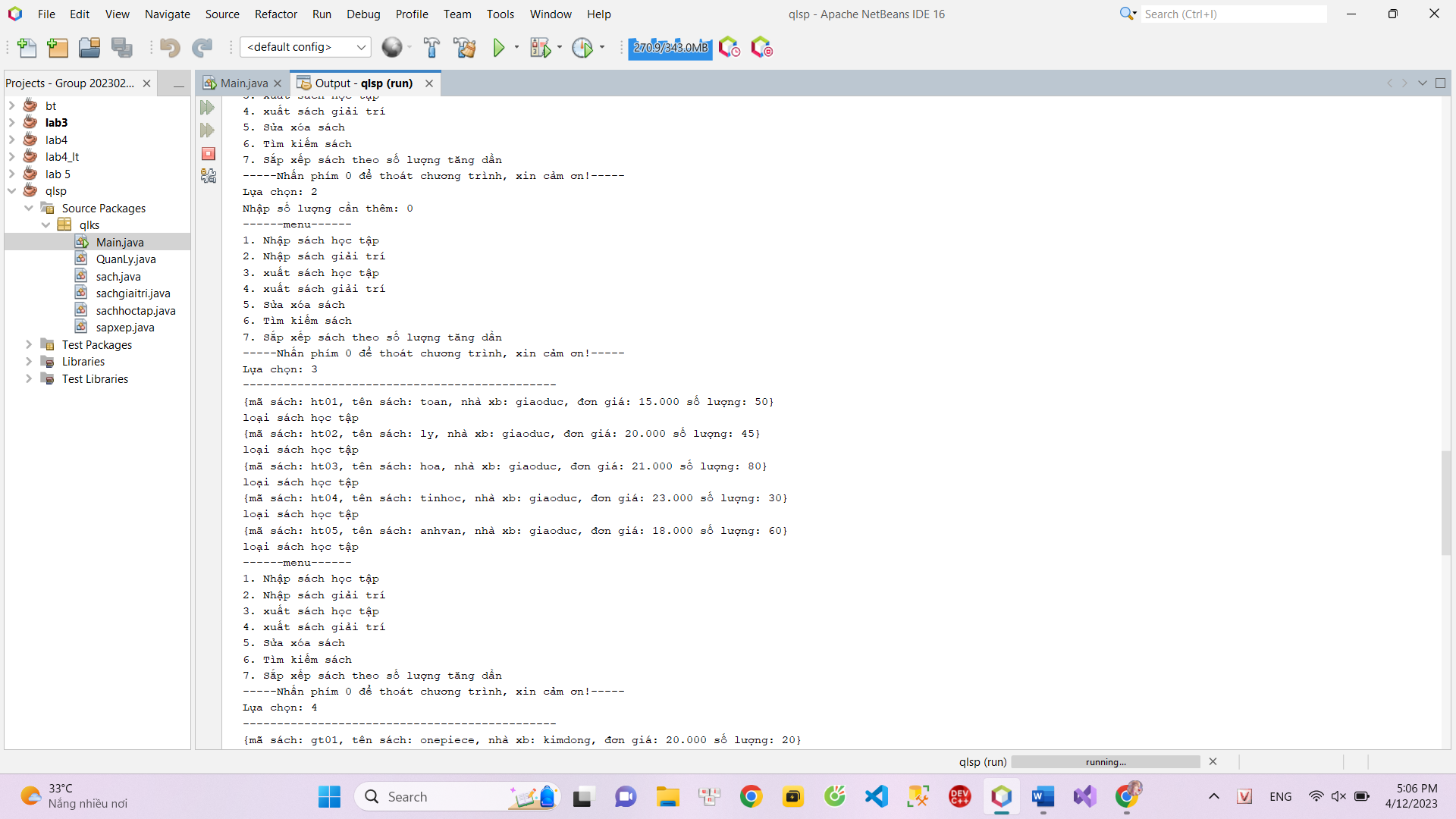
## **1. Nhập thông tin sách học tập**



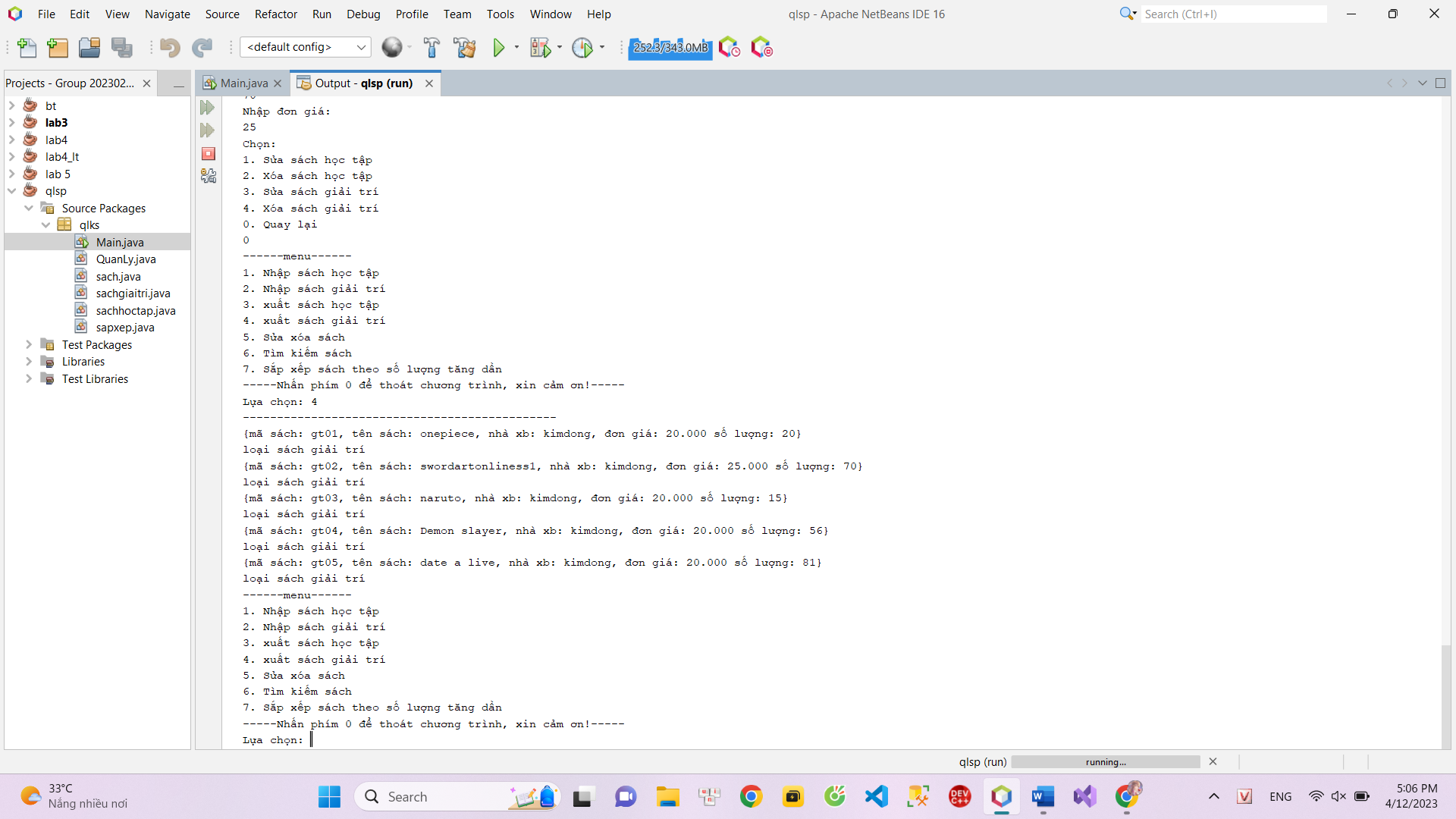
## **2. Nhập thông tin sách giải trí**



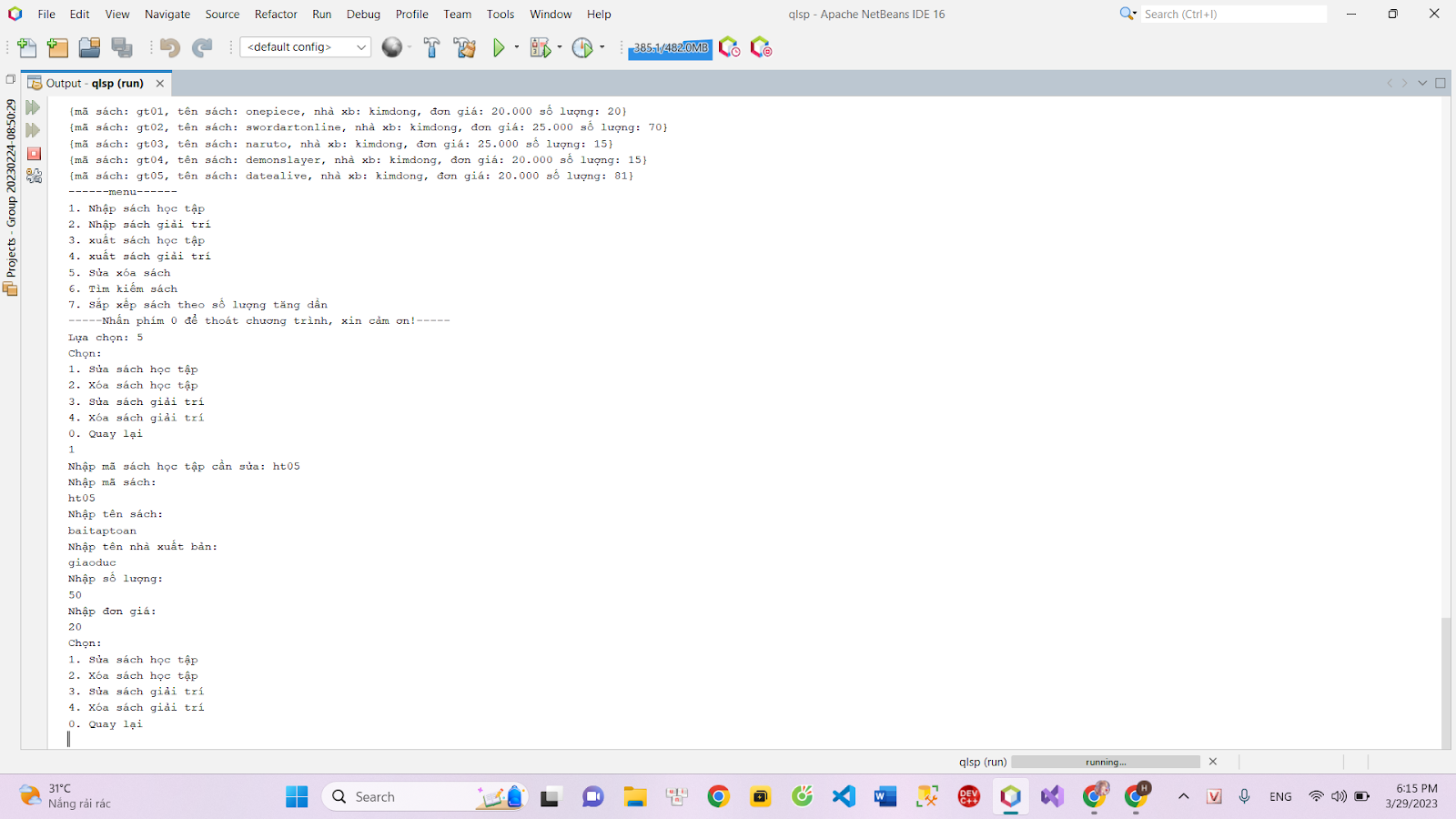
## **3. Hiện thông tin sách học tập**

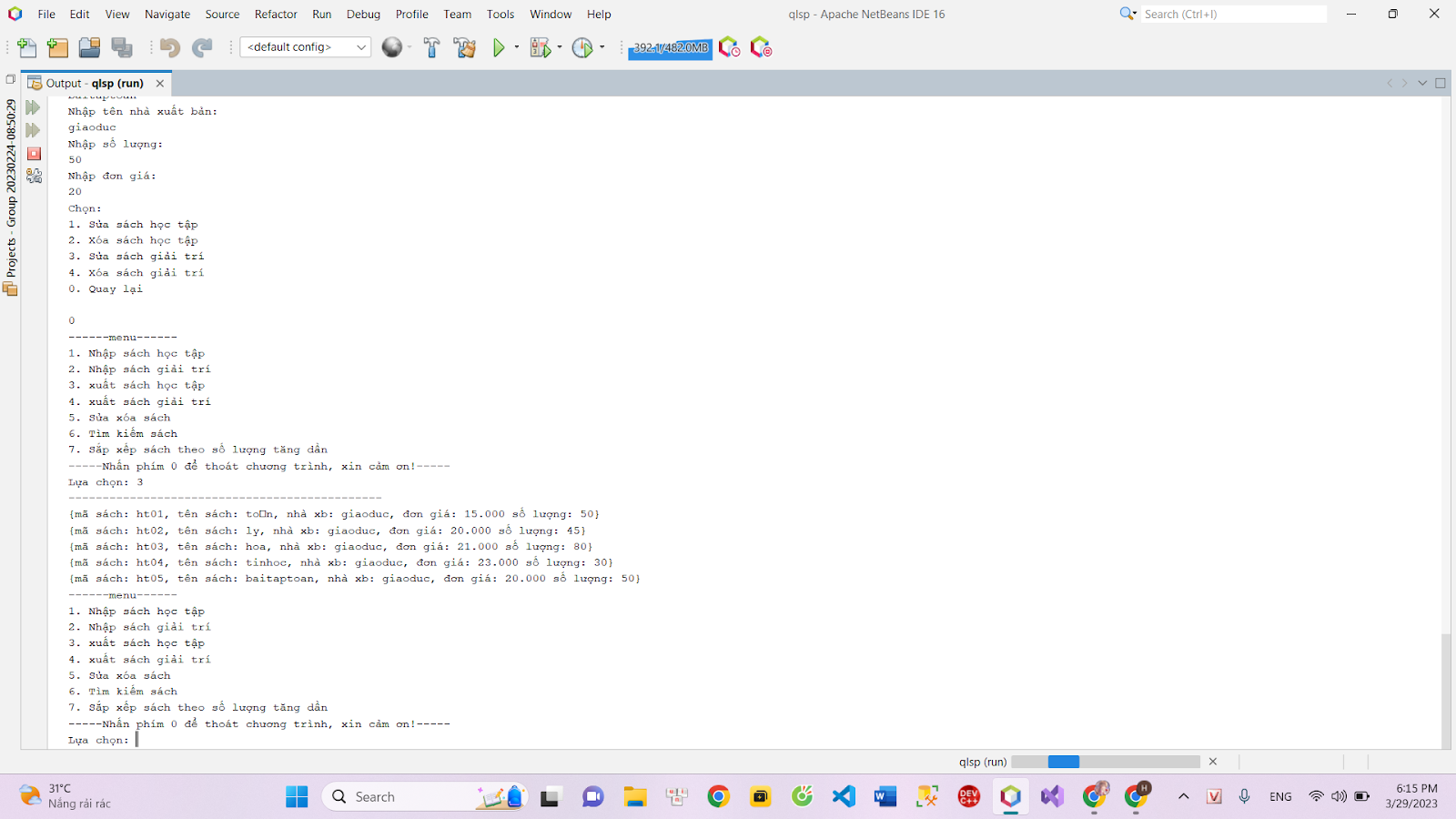


## **4. Hiện thông tin sách giải trí**

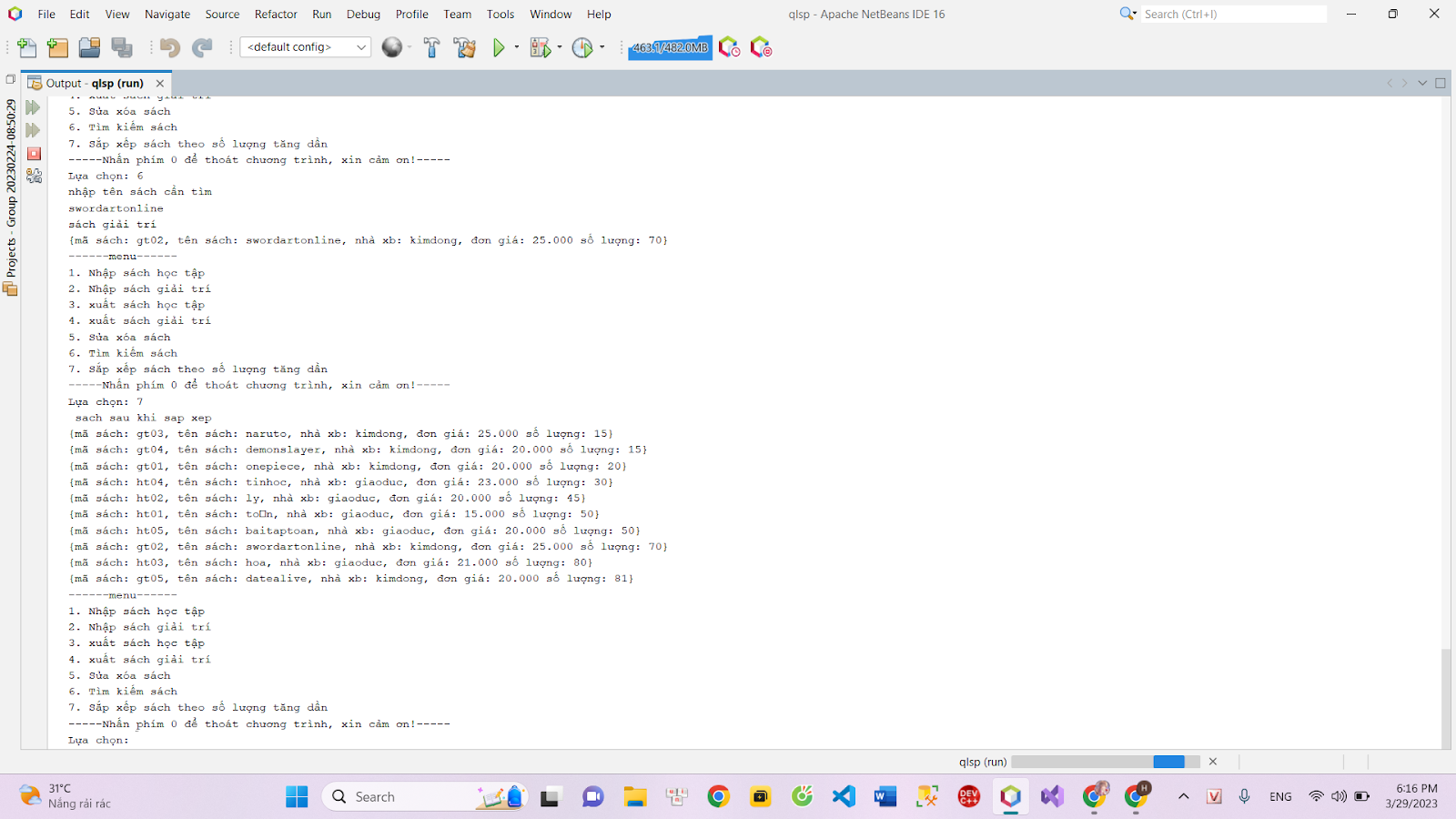


## **5. Chỉnh sửa thông tin( Thêm, Sửa, Xóa)**





## **6. Sắp xếp danh sách sách theo số lượng tăng dần**



## **7. Hiển thị sách cần tìm**

